



## City Research Online

### City, University of London Institutional Repository

---

**Citation:** Petropoulou, S. (2006). Adjoint-based geometry optimisation with applications to automotive fuel injector nozzles. (Unpublished Doctoral thesis, City University London)

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

---

**Permanent repository link:** <https://openaccess.city.ac.uk/id/eprint/8492/>

**Link to published version:**

**Copyright:** City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

**Reuse:** Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

# **Adjoint-Based Geometry Optimisation with Applications to Automotive Fuel Injector Nozzles**

**Stamatina Petropoulou Dipl (Maths) MSc (Eng)**

Thesis submitted in partial fulfilment of the requirements for the degree of  
Doctor of Philosophy

June 2006

Στο μπαμπά μου,  
που με μύησε στον κόσμο των μαθηματικών

“Σε οποιαδήποτε συγκεκριμένη θεωρία, περιέχεται τόση πραγματική  
επιστήμη όσα είναι τα μαθηματικά της”

“In any special doctrine of nature there can be only as much proper science  
as there is mathematics therein”

Immanuel Kant



# TABLE OF CONTENTS

<b>TABLE OF CONTENTS .....</b>	<b>4</b>
<b>LIST OF FIGURES .....</b>	<b>8</b>
<b>LIST OF TABLES .....</b>	<b>12</b>
<b>ACKNOWLEDGEMENTS.....</b>	<b>13</b>
<b>DECLARATION .....</b>	<b>15</b>
<b>ABSTRACT .....</b>	<b>16</b>
<b>NOMENCLATURE .....</b>	<b>17</b>
<b>CHAPTER 1 INTRODUCTION .....</b>	<b>21</b>
1.1 Motivation and general overview of optimisation methods.....	21
1.2 Present Contribution .....	23
1.3 Thesis Outline .....	25
<b>CHAPTER 2 LITERATURE REVIEW.....</b>	<b>27</b>
2.1 Introduction.....	27
2.2 General overview on automatic shape design methods .....	27
2.2.1 Inverse Design methods .....	29
2.2.2 Optimisation methods .....	30
2.2.3 Global vs. local methods.....	32
2.2.4 Gradient-based optimisation .....	35
2.2.5 Optimisation methods based on evolutionary computation algorithms.....	41
2.3 Adjoint-based optimisation .....	43
2.3.1 The discrete and the continuous approach .....	44
2.3.2 Properties of the adjoint solutions.....	49
2.3.3 Applications of the adjoint – based optimisation.....	53
2.3.4 Other applications of the adjoint method .....	60
2.4 Cavitation and other related studies in engine design.....	62
2.4.1 Work in internal flows and engine components optimisation.....	67
2.5 Background .....	70
2.6 Conclusions.....	74
<b>CHAPTER 3 FORMULATION OF THE OPTIMISATION PROBLEM.....</b>	<b>76</b>
3.1 Introduction.....	76

3.2 Mathematical Formulation of General Optimisation Problem.....	77
3.3 Optimisation Procedure.....	78
3.4 Formulating an objective function .....	81
3.5 Geometry Parameterisation.....	85
3.6 Grid generation.....	95
3.7 Governing equations for fluid flows .....	97
3.7.1 Turbulence Modelling.....	98
3.8 Adjoint Equations Analysis .....	99
3.8.1 Computing the gradients .....	99
3.8.2 Formulation of adjoint equations .....	100
3.8.3 Optimisation Procedure.....	101
3.8.4 The adjoint equations .....	102
3.8.5 Final form of the adjoint system of equations.....	111
3.9 Boundary conditions .....	112
3.9.1 Target pressure distribution .....	113
3.9.2 Given $C_d$ value.....	114
3.10 Formulation of gradients.....	117
3.11 Updating the Design Variables .....	119
3.12 Summary .....	120
<b>CHAPTER 4 DISCRETISATION AND SOLUTION OF THE FLOW AND ADJOINT EQUATIONS</b> .....	<b>121</b>
4.1 Introduction.....	121
4.2 The GFS code for computing the flow field .....	121
4.3 The Generic Transport Equation.....	122
4.4 Calculation of basic algebraic expressions .....	123
4.4.1 Face Interpolation.....	124
4.4.2 Differential Operators .....	126
4.4.3 Time derivative and Temporal Approximation.....	127
4.4.4 Convection Term.....	129
4.4.5 Diffusion Term.....	129
4.4.6 Source Term .....	129
4.4.7 Boundary Conditions .....	130
4.4.8 System of Linear Algebraic Equations .....	132
4.5 Discretisation of Single Phase Model Equations .....	134
4.5.1 Continuity Equation.....	134

4.5.2 Momentum Equation.....	134
4.5.3 Pressure Equation.....	137
4.5.4 Boundary Conditions .....	139
4.5.5 Sequence of Solution.....	142
4.5.6 Pressure-velocity coupling with PISO .....	143
4.5.7 Solution procedure of the governing equations system .....	143
4.6 Solution of the adjoint system of equations .....	144
4.6.1 Adjoint ‘Continuity’ Equation .....	144
4.6.2 Adjoint ‘Momentum’ Equation.....	145
4.6.3 Pressure correction for $\Psi_p$ .....	146
4.6.4 Boundary conditions .....	148
4.7 Summary .....	148
<b>CHAPTER 5 METHOD VALIDATION &amp; PARAMETRIC STUDIES.....</b>	<b>150</b>
5.1 Introduction.....	150
5.2 Reference Nozzle geometry for parametric studies .....	150
5.3 Effect of numerical parameters .....	156
5.3.1 Effect of Grid .....	156
5.3.2 Effect of flow and adjoint solution iterations - continuation .....	160
5.4 Effect of optimisation parameters .....	163
5.4.1 Effect of Steepest descent step size.....	163
5.4.2 Effect of different initial geometries.....	164
5.4.3 Effect of different target pressure distributions .....	166
5.5 Inverse design of 6-hole VCO Nozzle .....	168
5.6 Inverse design of an isolated airfoil .....	174
5.7 Summary .....	178
<b>CHAPTER 6 METHOD APPLICATIONS.....</b>	<b>181</b>
6.1 Introduction.....	181
6.2 Overview of test cases investigated .....	181
6.3 Axisymmetric single-hole nozzle.....	184
6.3.1 Parametric study of the effect of the design parameters .....	184
Effect of conicity.....	185
Effect of radius of curvature .....	187
6.3.2 Discharge coefficient maximisation of the 2-D axisymmetric single-hole nozzle .....	189

6.4 Multi-hole VCO NS nozzle.....	190
6.4.1 Effect of different parameters .....	191
6.4.2 Discharge coefficient maximisation of multi-hole VCO NS nozzle.....	193
6.5 Multi-hole VCO WS nozzle.....	195
6.5.1 Effect of design parameters.....	197
6.5.2 Effect of needle lift.....	199
6.6 Mini-sac Nozzle .....	202
6.6.1 Effect of geometrical constraints .....	204
6.7 Summary .....	207
<b>CHAPTER 7 CONCLUSIONS &amp; RECOMMENDATIONS FOR FUTURE WORK .....</b>	<b>209</b>
7.1 Overview .....	209
7.2 Conclusions .....	210
7.3 Recommendations for Future Work.....	213
<b>REFERENCES.....</b>	<b>215</b>

# LIST OF FIGURES

Figure 2.1: Mach distribution of EUROSUP in zero angle of attack and $M_\infty = 2$ .	72
Figure 2.2: The initial NURBS parameterisation of EUROSUP	73
Figure 2.3: Derived profile shape vs. the original (left) pressure distribution vs. the experimental one (right) of the <i>RAE2822</i> airfoil	73
Figure 2.4: Derived profile shape vs. the original (left) pressure distribution vs. the target one (right) of the <i>RAE2822</i> airfoil	73
Figure 2.5: Convergence history of the drag coefficient (left), Convergence history of the inverse design case (right).	73
Figure 3.1: Parameterisation of initial geometry using Bezier polynomials.	89
Figure 3.2: Parameterisation used for 2-D axisymmetric single hole nozzle redesign	91
Figure 3.3: Linear smoother for 2-D parameterisation	92
Figure 3.4: Parameters that affect the discharge coefficient	93
Figure 3.5: Example of a 2-D zone and representation of the coordinate system	94
Figure 3.6: Parameterised surface of a symmetric 3-D nozzle hole	95
Figure 3.7: Initial and modified nozzle mesh using the developed parameterisation and grid modification tools	96
Figure 4.1: Example of control volume	123
Figure 4.2: Control volume with a boundary face. The red line depicts the “pseudo-cell”	130
Figure 5.1: Axisymmetric 3-D nozzle for I-Level project (by courtesy of [216])	151
Figure 5.2: Real-size single-hole nozzles with varying degrees of hydro-grinding	151
Figure 5.3: Single-hole nozzles for (a) 0% and (b) 9.5% hydro-grinding. The calculated void fraction distribution is shown for comparison with the CCD images, taken from [180].	152
Figure 5.4: Predicted nozzle hole exit blockage	152
Figure 5.5: Predicted discharge coefficient as function of inlet radius of curvature and cavitation number	153
Figure 5.6: Example of parameterised I-Level nozzle	154
Figure 5.7: Initial and Target pressure fields of the axisymmetric nozzle	155
Figure 5.8: Three grids used for the grid effect parametric study	157

Figure 5.9: Pressure fields for the initial and target geometries using the three different grids .....	158
Figure 5.10: Initial and targeted nozzle designs and dimensionless (CN) pressure vs. model predictions after convergence for the three grids .....	159
Figure 5.11: Convergence history of the cost function for the reference, more iterations and no-continuation cases .....	160
Figure 5.12: Convergence history of the cost function for the reference, more iterations and no-continuation cases .....	161
Figure 5.13: Target dimensionless cavitation number (CN) corresponding to an initial geometry and model predictions after convergence, for (a) reference and (b) more iterations case .....	162
Figure 5.14: Initial, target and predicted nozzle shapes for (a) reference and (b) more iterations case .....	162
Figure 5.15: Convergence history of reference geometry with effect of step size .....	163
Figure 5.16: Geometry, pressure field and distribution for the four different initials ..	165
Figure 5.17: Convergence history of validation case using different initial geometries .....	165
Figure 5.18: Predicted vs. target geometries and pressure distributions for the four cases .....	166
Figure 5.19: Convergence history of validation case using different target pressure distributions .....	167
Figure 5.20: Geometry, pressure field and distribution predicted (black) vs. target (red) for the three different target cases. Initial (blue) geometry and pressure distribution.....	168
Figure 5.21: 3-D VCO 6-hole Nozzle geometry and test case simulated .....	169
Figure 5.22: The three design parameters used for the reproduction of the 3-D geometries .....	170
Figure 5.23: Pressure distribution where negative pressure is depicted at the hole entry and flow field along the hole plane of symmetry of the (a) initial geometry and the (b) target geometry .....	170
Figure 5.24: Cross sections at four locations parallel to the hole plane of symmetry, showing the initial geometry, the target geometry, the predicted geometry after 5 optimisation cycles and converged .....	171

Figure 5.25: Predicted vs. target and initial geometries and pressure distributions for the 3-D validation case in three ‘zones’ (a) upper, (b) lower and (c) middle .....	172
Figure 5.26: Convergence history of the cost function for the 3-D test case.....	173
Figure 5.27: Parameterisation of initial geometry using Bezier polynomials.....	174
Figure 5.28: Flow field around <i>NACA0012</i> airfoil (upper left) u-velocity, (upper right) v-velocity, (lower) pressure distributions.....	175
Figure 5.29: Adjoint variables around <i>NACA0012</i> airfoil (upper left) $\Psi_1$ , (upper right) $\Psi_2$ , (lower) $\Psi_p$ .....	176
Figure 5.30: Derived pressure distribution and profile shape vs. target <i>NACA0012</i> airfoil first case (upper) geometries, (lower) pressure distributions .....	177
Figure 5.31: Derived pressure distribution and profile shape vs. target <i>NACA0012</i> airfoil second case (upper) geometries, (lower) pressure distributions..	177
Figure 5.32: Convergence history of the geometry difference for the two test cases investigated.....	178
Figure 6.1: Details of erosion in nozzle due to (left) hole cavitation and (right) string cavitation .....	182
Figure 6.2: Half CAD model of axisymmetric 0%HE nozzle for I-Level project.....	184
Figure 6.3: Initial and optimised boundary of the axisymmetric nozzle for the three cases using conicity as design parameter .....	185
Figure 6.4: History of the discharge coefficient and cost function for the three test cases where only the conicity is used as design parameter.....	186
Figure 6.5: Initial vs. predicted for the three cases investigated using the conicity as design parameter where blue area indicates negative pressure .....	186
Figure 6.6: Initial and optimised boundary of the axisymmetric nozzle for the three cases using radius of curvature as design parameter .....	187
Figure 6.7: History of the discharge coefficient and cost function for the three test cases where only the radius of curvature is used as design parameter .....	188
Figure 6.8: Initial (blue line) vs. predicted (contour) for the three cases investigated using the radius of curvature as design parameter where blue area indicates negative pressure .....	188
Figure 6.9: History of the discharge coefficient and cost function for the optimisation case .....	189
Figure 6.10: Initial vs. optimised boundary of the axisymmetric nozzle after convergence of the cost function.....	189

Figure 6.11: Effect of different parameters of the VCO NS nozzle in the $C_d$ maximisation case .....	191
Figure 6.12: Variation of the discharge coefficient with respect to the variation of each parameter .....	192
Figure 6.13: History of the discharge coefficient and cost function for the 6-hole VCO NS nozzle case.....	193
Figure 6.14: Initial and optimised pressure contour of the VCO NS nozzle after convergence of the cost function.....	194
Figure 6.15: Streamlines and iso-surface of negative pressure of the (a) initial, (b) third cycle, (c) tenth cycle and (d) predicted geometries for the Multi-hole VCO NS nozzle case .....	195
Figure 6.16: CAD model of 6-hole VCO WS nozzle .....	196
Figure 6.17: Results from unsteady cavitating flow when (a) cavitation is initiated and (b) when it reaches the exit of the hole for the VCO WS nozzle .....	196
Figure 6.18: Effect of different parameters of the VCO WS nozzle in the $C_d$ maximisation case .....	198
Figure 6.19: History of the discharge coefficient and cost function for the VCO WS nozzle in two different needle lifts .....	200
Figure 6.20: Initial and optimised pressure contour of the VCO WS nozzle in 250 $\mu$ m (upper) and 20 $\mu$ m (lower) needle lift after convergence of the cost function.....	201
Figure 6.21: Streamlines and iso-surface of negative pressure of the (a) initial, (b) fifth cycle, (c) fourteenth cycle and (d) predicted geometries for the VCO WS nozzle case in high lift conditions .....	202
Figure 6.22: Half of one hole of the mini-sac 6-hole nozzle geometry .....	203
Figure 6.23: Results for cavitating flow the mini-sac multi-hole nozzle.....	203
Figure 6.24: History of the discharge coefficient for the mini-sac nozzle case without constraints.....	204
Figure 6.25: Initial (left) and optimised pressure contour of the mini-sac nozzle with constraint conicity (middle) and free parameters (right) needle lift after convergence of the cost function.....	205
Figure 6.26: Streamlines and iso-surface of negative pressure of the (a) initial, (b) fourth cycle, (c) eighth cycle and (d) predicted geometries for the mini-sac nozzle case.....	206



## LIST OF TABLES

Table 2.1: Historical summary of the design methods.....	32
Table 3.1: Coefficients in the standard $k-\epsilon$ model.....	99
Table 4.1: Summary of boundary conditions for each equation and boundary type ....	142
Table 4.2: Numerical solution procedure.....	144
Table 5.1: Geometrical details of the I-Level nozzles .....	153
Table 5.2: Properties of the I-Level project Diesel fuel.....	155
Table 5.3: Computational cell densities for 2-D grid effect study.....	156
Table 5.4: Cases showing the effect of different initial geometries.....	164
Table 5.5: Cases showing the effect of different target pressure distributions .....	166
Table 5.6: Initial and target geometry design parameters .....	169
Table 5.7: Comparison of CPU time required for adjoint optimisation.....	173
Table 6.1: Description of optimisation test cases.....	183
Table 6.2: Cases and results for the effect of conicity .....	185
Table 6.3: Cases and results for the effect of radius of curvature.....	187
Table 6.4: Summary of results for the effect of the design parameters in the Multi-hole VC NS nozzle optimisation .....	192
Table 6.5: Results of the Multi-hole VCO NS nozzle after convergence of the cost function.....	194
Table 6.6: Summary of the effect of the design parameters in the Multi-hole VCO WS nozzle optimisation.....	199
Table 6.7: Summary of the effect of the needle lift in the Multi-hole VCO WS nozzle optimisation .....	200
Table 6.8: Results of the mini-sac nozzle for constraint and unconstraint optimisation .....	205

## ACKNOWLEDGEMENTS

It is not often that a person gets a second chance to complete something that was left unfinished. This chance was given to me, here in City University, from my supervisor Dr Manolis Gavaises whom I deeply thank for the opportunity as well as for the knowledge he shared with me in the automotive engineering area. Coming from an aeronautical environment his help has been really valuable.

I would also like to thank Prof. C. Arcoumanis for including me in his group and supporting me in any way he could during the time of my studies. I also thank him for providing me with the freedom to work on the subject of my choice.

Thanks are also owed to everybody in the CFD group for the help support and endless conversations. Especially to Dr A. Theodorakakos for providing the GFS code and helping me on the implementation of my part. Moreover, I would like to thank him and Fluid Research Co. for the availability of precious computational resources and other technical equipment. To Dr E. Giannadakis I owe a special “thank you” for having the patience to explain to me the concept of cavitation more than once. I would like to thank him and Mr D. Papoulias for providing me with the pictures of cavitating flows included in this thesis. From the colleague but also from the female point of view I especially thank the two other women of the group Miss S. Tonini and Miss M. Spathopoulou, for being there. It is always nice to have a girl talk when you are surrounded by men and especially engineers. I would like to thank those two for the work related discussions as well, although I must admit this spray thing I never understood. Thanks also go to Mr A. Adriotis, Mr G. Strotos mainly for the music (they know).

A big thank you goes also to my other friends and colleagues from the research groups at City for being around and for helping me in one way or the other: Mr A. Dhunput, Mr M. Hamid, Dr E. Karaiskos, Mr L. Liverani, Mr A. Marchi, Mr N. Mitroglou, Mr K. Mori, Mr T. Ous, Mr K. Wang, Mr M. Chester, Mr A. Metioui, Mr O. Pierce and Mr D. Guerrato.

A big part of the research for this thesis was done in NTUA. I would like to thank my advisor there Prof. Giannakoglou for introducing me to CFD and optimisation as well as Dr Koubogiannis and Dr Giotis for teaching me CFD and programming. Last but not least, I would like to thank my ex-colleague and dear friend Dr. Theodora Pap-

pou for all the help and support she has given me at a scientific and personal level and all the great ideas she shared with me during this research.

On a personal level I thank my family and my friends for being close to me although I am so far from them. Especially I want to thank N. Charalambous because without him I would have never come to London. Finally, I would like to give also a big, warm thanks to N. Free. Although, this thesis took so long and I know him for so little, his support has been very helpful during this last year.

Stamatina Petropoulou

July 2006

## **DECLARATION**

The author grants power of discretion to the University Librarian to allow the thesis to be copied in whole or in parts without further reference to the author. This permission covers only single copies made for study purposes, subject to normal conditions of acknowledgement.

## ABSTRACT

Methods of Computational Fluid Dynamics (CFD) have matured, over the last 30 years, to a stage where it is possible to gain substantial insight into fluid flow processes of engineering relevance. However, the motives of fluid dynamic engineers typically go well beyond the level of improved understanding, to the pragmatic aim of improving the performance of the engineering systems in consideration. It is in recognition of these circumstances that the present thesis investigates the use of automated design optimisation methodologies in order to extend the power of CFD as an engineering design tool. Optimum design problems require the merit or performance of designs to be measured explicitly in terms of an objective function. At the same time, it may be required that one or more constraints should be satisfied. To describe allowable variations in design, shape parameterisation using basic geometric entities such as straight lines and arcs is employed.

Taking advantage of previous experience in the research group concerning cavitating flows, a fully automated method for nozzle design/optimisation was developed. The optimisation is performed by means of discharge coefficient ( $C_d$ ) maximisation. The objective is to design nozzle hole shapes that maximise the nozzle  $C_d$  for a given basic nozzle geometry (i.e. needle and sac profile) and reduce or even eliminate the negative pressure region formed at the entry of the injection hole. The deterministic optimisation model was developed and implemented in the in-house RANS CFD code to provide nozzle shapes with pre-defined flow/performance characteristics. The required gradients are calculated using the continuous adjoint technique. A parameterisation scheme, suitable for nozzle design, was developed. The localised region around the hole inlet, where cavitation inception appears, is parameterised and modified during the optimisation procedure, while the rest of the nozzle remains unaffected. The parameters modifying the geometry are the radius of curvature and the diameter of the hole inlet or exit as well as the relative needle seat angle. The steepest descent method has been used to drive the calculated gradients to zero and update the design parameters.

For the validation of the model two representative inverse design cases have been selected. Studies showing the behaviour of the model according to different numerical and optimisation parameters are also presented. For the purpose of optimising the geometries, a cost function intended to maximise the discharge coefficient was defined. At the same time it serves the purpose of restructuring geometries which have controlled or eliminated cavitation inception in the hole entrance. This is identified in the steady-state mode by reduction of the volume of negative relative pressure appearing in the hole entrance. Results of cavitation control on some representative nozzle geometries show significant benefits gained by the use of the developed method. This is mainly because the developed model performs optimisation on numerous parametric combinations automatically. Results showed that, by using the proposed method, geometries with larger  $C_d$  values can be achieved and the cavitation inception can, in some cases, be completely eliminated. Cases where all the parameters were combined for redesign the geometry required less modification to predict larger  $C_d$  values than cases where each parameter was modified individually. This is an important result since manufacturers are seeking improvement in the performance of products resulting from the least geometry modifications.

# NOMENCLATURE

## Abbreviations

1-D	One-dimensional
2-D	Two-dimensional
3-D	Three-dimensional
AD	Automatic Differentiation
BD	Blended differencing
CAD	Computer aided design
CD	Central differencing
CCD	Charge coupled device
CFD	Computational fluid dynamics
CPU	Central processing unit
CT	Computed tomography
CV	Control volume
DSA	Design sensitivity analysis
I-Level	Injector Flow – Low Emission levels by engine modelling
LDV	Laser Doppler velocimetry
l.h.s.	Left-hand-side
NACA	National advisory committee for aeronautics
ODESSY	Optimum DESign System
PDE	Partial differential equation
UD	Upwind differencing
RAE	Royal Aircraft Establishment
r.h.s.	Right-hand-side
STD	Standard
TKE	Turbulent kinetic energy
VCO	Valve covering orifice
VNP	Volume of negative pressure
VOF	Volume of fluid

### Roman Symbols

<b>A</b>	Linear system matrix coefficient	-
	Projected area	$m^2$
<b>a</b>	Acceleration	$m/s^2$
<b>B</b>	Linear system source coefficient	-
<b>b</b>	Body forces per unit mass vector	$m/s^2$
<b>c</b>	Speed of sound	$m/s$
	Mass flux	
<b>C</b>	Bezier coefficients	$kg/s$
<b>d</b>	Vector between two cell centres	$m$
<b>d</b>	Injection hole diameter	$m$
<b>D</b>	Orthogonal contribution vector	$m$
<b>D</b>	Design parameters vector	-
<b>E</b>	Constant for wall-function approach	-
<b>f</b>	Interpolation factor	-
<b>F</b>	Volume flux	$m^3/s$
<b>F</b>	Force vector	$N$
<b>g</b>	Vector of gravitational acceleration	$m/s^2$
<b>I</b>	Cost function	-
<b>I</b>	Unit tensor	-
<b>J</b>	Constraint function	-
<b>k</b>	Turbulent kinetic energy	$m^2/s^2$
<b>k</b>	Non-orthogonal contribution vector	$m$
<b>L</b>	Injection hole length	$m$
<b>m</b>	Mass	$kg$
<b>p</b>	Pressure	$bar, Pa$
<b>P</b>	Turbulence production	$kg /m.s^3$
<b>q</b>	Symbolic term for discretised continuity equation	-
<b>S</b>	Outward pointing face area vector	-
<b>S</b>	Continuous phase source term	-
<b>T</b>	Stress tensor	$m^2/s$
<b>T</b>	Temperature	$K$
<b>t</b>	Time	$s$

$t$	Bezier variable	-
$\mathbf{u}$	Velocity vector	m/s
$u$	X-axis or axial velocity component	m/s
$v$	Y-axis or radial velocity component	m/s
$V$	Velocity magnitude	m/s
$V$	Volume	$m^3$
$w$	Z-axis velocity component	m/s
$X$	X-axis Bezier control point	-
$Y$	Y-axis Bezier control point	-
$y^+$	Dimensionless distance from the wall	-

### **Greek Symbols**

$\alpha$	Void fraction	-
$\beta$	Time integration scheme blending factor	-
$\gamma$	Polytrophic coefficient	-
$\delta$	Variation	-
$\varepsilon$	Turbulence dissipation rate	$m^2/s^3$
$\eta$	Steepest descent step size	-
$\kappa$	Von Karman's constant	-
$\lambda$	Relative air/fuel ratio	-
$\mu$	Dynamic viscosity	kg/m.s
$\nu$	Kinematic viscosity	$m^2/s$
$\zeta$	Under-relaxation factor	-
$\rho$	Density	kg/m <sup>3</sup>
$\sigma$	Surface tension	N/m
$\sigma$	Standard variation	-
$\varphi$	Arbitrary scalar or tensor	-
$\Psi$	Adjoint/costate variables	-
$\Psi$	Adjoint vector	-

### **Subscripts**

$Q_{adj}$	Q of adjoint
$Q_{aug}$	Q Augmented



$Q_{\text{back}}$	Q at downstream location
$Q_c$	Cost
$Q_d$	Molecular (for viscosity)
$Q_d$	Discharge
$Q_f$	Q at face f or face interpolation
$Q_e$	Q effective
$Q_i$	Q of $i=1, \dots, n$
$Q_{\text{inj}}$	Q of injection (for velocity: inside the nozzle hole)
$Q_p$	Q related to the pressure or pressure gradient force
$Q_u$	Q related to the velocity vector
	Target Q
$Q_t$ or $Q_{\text{tar}}$	Eddy (for viscosity)

### **Superscripts & Oversymbols**

$Q^\perp$ or $\nabla^\perp Q$	Orthogonal component of Q or of its gradient
$Q^t$	Q at time t
$\overline{Q}$	Time or ensemble average of Q Known/target value of Q
$\vec{Q}$	Vector
$\dot{Q}$	Derivative with respect to the design variables
$[Q]$	Volume average of Q

### **Non-dimensional numbers**

$C_d$	Orifice discharge coefficient
CN	Cavitation number
Re	Reynolds number

# *Chapter 1*

## *INTRODUCTION*

### **1.1 MOTIVATION AND GENERAL OVERVIEW OF OPTIMISATION METHODS**

Cavitation is the formation of a two-phase mixture of vapour/gas and liquid at the point where the local pressure of the liquid drops below a threshold value, a value usually considered equal to the vapour pressure. However, theoretically speaking fluids can withstand large amounts of tension – meaning pressures below the thermodynamic vapour pressure value at a given temperature – before phase change from liquid to vapour is induced [1]. This phenomenon appears in the nozzle of automotive fuel injectors in two forms. The first type of cavitation initiates inside the nozzle hole at the top of its entrance (*geometric cavitation*) and is due to flow separation at the corner causing local pressure drop. Sometimes, cavitation erosion appears in this area. Cavitation is inherently a multiphase flow. Single-phase CFD technology is considered to be at such a mature level that single-phase predictions can be well trusted and used by the industry for the solution of complex problems and the development of new products. According to single-phase incompressible flow calculations, the area of geometric cavitation inception can be correlated with the area where the calculated pressure takes values below the vapour pressure of the flowing liquid.

The second form of cavitation is referred to as string cavitation; the root of this two-phase flow effect is still not well documented. Recent experimental evidence suggests that string cavitation is likely to appear due to vapour entrapment into the core of the transient vortices forming inside the sac volume upstream of the injection hole. In the present study only the reduction of the geometrically-induced cavitation is considered. For most hydraulic equipment cavitation is not desirable, since its mere existence results in efficiency and power loss, mechanical wear which can be catastrophic in many cases, and noise [2, 3]. Although the effect of cavitation on the spray and the overall engine performance is still under investigation, it is well recognised that uncontrolled cavitation induces spray-to-spray and injection-to-injection variations which, in turn, increase en-

engine exhaust pollutant emissions. On the other hand, controlled cavitation can improve atomisation of the fuel which can lead to improved mixing and combustion.

Controlling cavitation greatly depends on nozzle geometry and operating conditions. In most diesel injectors operating at high pressure, cavitation initiates at the entrance of the nozzle hole. With any manifestation of cavitation, the objective is primarily the same as in single-phase flows, to increase the localised minimum pressure. Of course, locally increasing the minimum pressure may induce cavitation elsewhere or in another form that was previously not an issue. Hence, the problem of controlling cavitation is not trivial.

In the automotive industry, experiments by trial and error are still the main design tool. However, it's not at all certain that repeated tests in an interactive design and analysis procedure can lead to a truly optimum design. The use of an automated optimisation method allows designs to be rapidly evaluated while directions of improvement are identified as well. Possession of such techniques that result in a faster design cycle gives a crucial advantage in a competitive environment.

The evolution of Computational Fluid Dynamics (CFD), especially in the last decade, has dramatically altered the optimal design of vehicles either in the sense of aerodynamic or engine component design. This is due to the tremendous advances in CFD and the development of increasingly faster and larger-memory computers, which make possible the use of CFD in shape design. In the same period there has been a rapid development of a powerful collection of algorithms for gradient based optimisation.

Shape optimisation is a procedure where the geometry of interest is modified to fit certain flow characteristics. Introduction of the flow characteristics into the problem is achieved by the definition of a so-called 'cost' function. The geometry is modified according to a set of parameters defined by the user. In deterministic optimisation the cost function is minimised by driving its gradients to zero with the use of a descent method. Using direct methods for the calculation of the required gradients in the case of nozzle design is expensive in CPU time due to the large number of design parameters. For this reason, methods such as the finite differences that were most commonly used in the past are not popular anymore and the development of adjoint methods for the calculation of the gradient is recommended. It is appropriate at this point to mention earlier work in

aerodynamics based on control-theory by Pironneau [4, 5] and Jameson [6] who is considered to be the father of the adjoint method. During the last decade several applications of the adjoint method were brought to light [7-10]. So far, most work on the adjoint method in CFD shape optimisation has focused on aerodynamic design, which involves very simple boundary conditions, especially those in the far field. In this case the corresponding boundary conditions for the adjoint equations are easier to formulate. Relevant research has been conducted in engine component optimisation but for very simple geometries.

It must be stated that computational design optimisation in no way is suggested to replace present design methodologies. The intent of design optimisation is not to do preliminary or primary design, but to be used to supplement the design process, by making improvements on initial designs in areas where no design methodologies exist. When starting from a fairly good design, formal optimisation is most effective to locate non-intuitive designs.

## 1.2 PRESENT CONTRIBUTION

The objective of this research programme was the development of an automated mathematical methodology for designing nozzle geometries with specified flow characteristics. More specifically, the goal was to propose a design tool for automotive diesel injector nozzles which eliminate or produced ‘controlled’ cavitation resulting to more specified spray structures. In the latter case, this implies designs where the integral amount of cavitation reaching the hole exit can be one of the desired flow parameters.

The optimisation method developed for the needs of this study is based on the adjoint formulation of the flow equations. The proposed method is a deterministic optimisation method and requires gradient information, i.e. sensitivity derivatives. The adjoint formulation is a method based in control theory and the theory of Lagrange multipliers that provides the gradients of the cost function based. The flow solution is obtained from a steady-state incompressible Navier-Stokes equations flow solver, provided that such a solution exists. Moreover, the adjoint formulation is carried out by incorporating the flow equations, in the form of constraints, into the cost function; thus the augmented form of the cost function is provided. This can be accomplished through the use of Lagrange multipliers, namely the adjoint or costate variables. The gradients of the cost

function depend on the flow field and the adjoint variable distribution along the wall to be designed. So, through the efficient solution of the flow and the adjoint equations the values of the sensitivity derivatives are provided. This way the gradients for each design parameter are calculated and driven to zero in order to direct the geometry towards an optimum. The continuous approach is adopted in this study so that the resulting adjoint formulation, optimisation algorithm, and overall code can be used in conjunction with any CFD code. The adjoint method has been chosen for optimisation due to its accuracy and fast convergence. In addition, it provides the gradients of the cost function in a way so that the computational effort required for this calculation is independent of the number of the design variables. This is an important advantage for nozzle re-design cases where the required parameters describing the geometry and affecting cavitation may be numerous. Nevertheless, the possibility of getting trapped into local minima exists as in every gradient-based method.

In order to setup the optimisation problem using the aforementioned method, some additional elements are required. The cost functions were defined in such a way that, by driving them to zero, geometries satisfying different flow characteristics were provided. The cost function is the means to quantify the optimum solution; as such it is really important to define it appropriately for each problem. For the purpose of controlling cavitation a cost function, aiming to maximise the discharge coefficient, was used. At the same time, maximisation of the discharge coefficient led to geometries which have controlled or eliminated cavitation inception in the hole entrance. This is identified in the steady-state mode by reduction of the volume of negative pressure appearing in the hole entrance. A crucial part of the development process was the setup of the aforementioned problem for the  $C_d$  maximisation. It should be noted that, this case was challenging with respect to the adjoint analysis. The complication when applying this design approach arises from the fact that the cost function is defined at the hole exit, whilst the gradients of the geometry under modification are calculated at the hole inlet. To overcome this problem, appropriate boundary conditions were assigned to the wall boundary under modification.

A parameterisation scheme suitable for the design of automotive injector nozzles was developed and implemented in the code. The points that control the flow characteristics that we want to optimise were used as parameters. A linear polynomial smoother was incorporated in the parameterisation scheme to avoid discontinuity of the nozzle geome-

tries and thus, production of poor quality grids. A grid modification tool was used to map the modified geometry onto the initial grid that was provided by a commercial program.

The final step in completing the iterative optimisation process is the steepest descent method that drives the calculated gradients to zero and provides the design parameters for the modification of the geometry towards the optimum solution.

In order to reduce the overall computational cost of the method, continuation in the flow calculation using the solution of the previous optimisation cycle (previous geometry) has been used. This resulted in significant CPU time reduction (sometimes even 90%) compared to a flow calculation which is initialised for every optimisation cycle.

To investigate the performance of the proposed methodology, the effect of various numerical parameters of the flow and adjoint equations, as well as parameters related to the optimisation method, was explored. The validation of the model was carried out by means of inverse design with respect to a prescribed pressure distribution of a 2-D axisymmetric and a 3-D multi-hole VCO nozzle. For the case of  $C_d$  maximisation several test cases, both 2-D and 3-D, were investigated. Parametric studies of the effect of the different design parameters on the optimum geometry were also performed.

The developed optimisation method for diesel injector nozzle design provides information regarding the direction of improvement and is effective in locating non-intuitive designs. In the case of  $C_d$  maximisation, even a slight modification of the design parameters can provide nozzles with reduced or eliminated cavitation (according to the specifications of the cost function and the design constraints). This is an important result since manufacturers are seeking improvement in the performance resulting from the least geometry modifications. Due to its automatic nature many cases can be easily investigated providing an insight into the sensitivity of the nozzle performance to changes of certain design parameters or combinations of them.

### 1.3 THESIS OUTLINE

The present thesis consists of seven chapters including the introductory *Chapter 1*, where the motivation and present contribution of this research work is outlined.

*Chapter 2* reviews publications in all the research areas that are relevant to shape optimisation methods and especially the adjoint method. The properties of the adjoint solutions are summarised along with other areas of the method's applications. Some information about the cavitation formation used for the development of the present method and work related to engine optimisation is also discussed.

In *Chapter 3* the proposed method is described and all the necessary steps for the optimisation are discussed in detail. Moreover the parameterisation developed for this study and the analysis of the adjoint formulation are also included in this chapter.

In *Chapter 4* the discretisation of the flow equations is described as the basis for the discretisation of the adjoint equations which is also discussed. The two systems of equations are cross-referenced and the corresponding similarities and differences are identified and described.

In *Chapter 5* the effect of various numerical parameters on the developed method is initially investigated followed by the effect of the parameters used for the optimisation process. The validation of the proposed method is also discussed in this chapter.

In *Chapter 6* various applications of the model are investigated and the corresponding results and conclusions drawn from the test cases are discussed in detail.

*Chapter 7* concludes the thesis with a summary of the most important points, in addition to the brief summaries in each chapter. Recommendations for further work in the area of nozzle shape optimisation are also presented in this chapter in order to extend the value of this contribution.

## *Chapter 2*

### *LITERATURE REVIEW*

#### **2.1 INTRODUCTION**

This chapter is a review of the published research in optimisation and other subjects related to this study. Special reference is given to the matters of shape optimisation and inverse design in CFD applications. In this area, work has been done more in aeronautical design and less in automotive which is the subject of the thesis. The optimisation method developed in this study is a gradient based method using continuous adjoint techniques for the calculation of the gradients. The different theories behind the method are cross referenced and discussed along with the suggestions of the author. The properties of the adjoint solution are also mentioned. Relevant studies on the topic of cavitation inception are briefly discussed especially the effect of shape changes on the cavitation inception. The few publications about engine component optimisation and cavitation control, mainly in very simple geometries are also referenced. Since the present study is a continuation of previous studies of the author in the field of aerodynamic optimisation some background work is indicated. The purpose of this chapter is to make an overview of the problem of shape optimisation and to connect the literature to the theoretical aspects of this thesis.

#### **2.2 GENERAL OVERVIEW ON AUTOMATIC SHAPE DESIGN METHODS**

The first part of the literature review is a detailed historical overview on the computational methods being used for shape design. The first ever developed inverse design method is described as well as how engineers turned to more efficient optimisation methods. The philology around global vs. local optimisation is also discussed.

Computers have had a twofold impact on the science of automotive engineering. On the one hand numerical simulation may be used to gain new insights into the physics of



complex flows. On the other hand computational methods can be used by engineers to predict the flow characteristics of alternative designs without the need to actually build them. Assuming that one has the ability to predict the performance, the question then arises of how to modify the design to improve this performance. The present study is addressing this question.

Prior to 1960 computational methods were hardly used in fluid flow analysis. The primary tool for the development of engine configurations was experiments based on trial and error. Shapes were tested and modifications selected in the light of pressure and force measurements together with flow visualisation techniques. Computational methods are now quite widely accepted as design tools by the automotive industry. This has been brought out by a combination of radical improvements in numerical algorithms and continuing advances in both speed and memory of computers.

If a computational method is to be useful in the design process, it must be based on a mathematical model which provides an appropriate representation of the significant features of the flow, such as cavitation, vortices and boundary layers. The method must also be robust, not liable to fail when parameters are varied, and it must be able to treat useful configurations, ultimately the complete engine. Finally, reasonable accuracy should be attainable at reasonable cost. Much progress has been made in these directions [11-21]. The most challenging viscous simulations are generally complicated by the need to allow for turbulence: while the Reynolds averaged equations can be solved by current methods, the results depend heavily on the choice of turbulence models. In particular it has been recognized that the designer generally has an idea of the kind of pressure distribution that will lead to the desired performance. Thus, it is useful to consider the problem of calculating the shape that will lead to a given pressure distribution. Such a shape does not necessarily exist, unless the pressure distribution satisfies certain constraints, and the problem must therefore be very carefully formulated.

All the above raised the need for developing more efficient and accurate methods for shape design in CFD. Methods for shape design in CFD can be classified into two major groups: inverse and optimisation. The distinction between the two is in how the design problem is formulated.

### 2.2.1 Inverse Design methods

In the inverse method, the shape-design problem is posed by assuming that certain flow characteristics, such as pressure, can be imposed on the domain boundary in a way that optimises the performance or some other desired property (flow characteristics). Once the problem is thus posed, the task is to find the shape or geometry that would predict the imposed distribution. Though this method can be highly efficient, it has two main drawbacks. The first is that the distribution imposed in the inverse problem formulation may not be physically realisable so that a solution may be impossible. The second drawback is that even if the distribution imposed is physically realisable, it may not be the optimal distribution.

The use of inverse methods to optimise aerodynamic shapes was pioneered by Lighthill [22] in the mid 1940s. He was the one who first studied the problem of designing a two dimensional profile to attain a desired pressure distribution. Lighthill solved this problem for the case of incompressible flow by conformal mapping of the profile to a unit circle. The speed over the profile is  $q = \varphi_\theta / h$  where  $\varphi$  is the potential for flow past a circle, and  $h$  is the modulus of the mapping function. The solution for  $\varphi$  is known for incompressible flow. Let  $q_d$  be the desired surface speed; then the surface value of  $h$  can be obtained by setting  $q = q_d$ , and since the mapping function is analytic, it is uniquely determined by the value of  $h$  on the boundary. A solution exists for a given speed  $q_\infty$  at infinity only if  $\frac{1}{2\pi} \oint q d\theta = q_\infty$  and there are additional constraints on  $q$  if the profile is required to be closed. Lighthill's method was extended to compressible flow by McFadden [23]. Starting with a given shape, and a corresponding mapping function  $h^{(0)}$ , the flow equations can be solved for the potential  $\varphi^{(0)}$ , which now depends on  $h^{(0)}$ . A new mapping function  $h^{(1)}$  is then determined by setting  $q = q_d$ , and the process is repeated. In the limiting case of zero Mach number the method reduces to Lighthill's method, and McFadden gives a proof that the iterations will converge for small Mach numbers. He also extended the method to treat transonic flow through the introduction of artificial viscosity to suppress the appearance of shock waves, which would cause the updated mapping function to be discontinuous. This difficulty can also be overcome by smoothing the changes in the mapping function. Such an approach allows the recovery of smooth profiles that generate flows containing shock waves, and it

has been used to design improved blade sections for propellers [24]. A related method for three-dimensional design was devised by Garabedian and McFadden [25]. In their scheme the steady potential flow solution is obtained by solving an artificial time dependent equation, and the surface is treated as a free boundary. This is shifted according to an auxiliary time dependent equation in such a way that the flow evolves toward the specified pressure distribution.

Another way to formulate the problem of designing a profile for a given pressure distribution is to integrate the corresponding surface speed to obtain the surface potential. The potential flow equation is then solved with a Dirichlet boundary condition, and a shape correction is determined from the calculated normal velocity through the surface. This approach was first tried by Tranen [26]. Volpe and Melnik [27] have shown how to allow for the constraints that must be satisfied by the pressure distribution if a solution exists. The same idea has been used by Henne [28] for three-dimensional design calculations. The hodograph transformation offers an alternative approach to the design of airfoils in transonic flows. Garabedian and Korn [29] achieved a striking success in the design of airfoils to produce shock-free transonic flows by using the method of complex characteristics to solve the equations in the hodograph plane. Another design procedure has been proposed by Giles, Drela and Thompkins [30], who write the two-dimensional Euler equations for inviscid flow in a streamline coordinate system, and use a Newton iteration. An option is then provided to treat the surface coordinates as unknowns, while the pressure is fixed.

### 2.2.2 Optimisation methods

The inverse method is severely restrictive because it depends on the experience and knowledge of the designer to establish the desirable target velocity or pressure distributions. That's why the engineers moved to more sophisticated optimisation methods.

In the case of optimisation, the shape-design problem is posed as a minimisation problem of an objective function, subject to constraints on geometry and/or flow conditions. Optimisation algorithms are iterative. They begin with an initial guess of the optimal values of the variables and generate a sequence of improved estimates until they reach a solution. The strategy used to move from one iterative to the next distinguishes one algorithm from another. Most strategies make use of the values of the objective (cost)

function  $I_C$ , the constraints and possibly the first and the second derivative of these functions.

Optimisation methods, pioneered by Hicks et al [31], are more attractive since these methods can handle a large class of design problems, including those classified as inverse problems. Hicks and Henne [32] have explored the possibility of meeting desired design objectives by using constrained optimisation. The configuration is specified by a set of parameters, and any suitable computer programme for flow analysis can be used to evaluate the aerodynamic characteristics. The optimisation method then selects values of these parameters that maximise some criterion of merit, such as the lift-to-drag ratio, subject to other constraints, for example the required wing thickness and volume. In principle this method allows the designer to specify any reasonable design objectives. The method becomes extremely expensive, however, as the number of parameters is increased, its successful application in practice depends heavily on the choice of a parametric representation of the configuration.

Jameson [6] was the first to propose that there are benefits in regarding the design problem as a control problem, the control being the shape of the boundary. A variety of alternative formulations of the design problem can then be treated systematically by using the mathematical theory for control of systems governed by partial differential equations [33]. He supposed that the boundary is defined by a function  $f(x)$ , where  $x$  is the position vector. As in the case of optimisation theory applied to the design problem, the desired objective is specified by a cost function  $I_C$ , which may, for example, measure the deviation from a desired surface pressure distribution, but could also represent other measures of performance such as lift and drag. The introduction of a cost function has the advantage that if the objective is unattainable, it is still possible to find a minimum of the cost function. Jameson's application of control theory can be summarised as follows. A variation in the control  $\delta f$  leads to a variation  $\delta I_C$  in the cost. It is shown in [6] that  $\delta I_C$  can be expressed to first order as an inner product of a gradient function  $g$  with  $\delta f$  such as  $\delta I_C = (g, \delta f)$  where  $g$  is independent of the particular variation  $\delta f$  in the control, and can be determined by solving an adjoint equation. By choosing a sufficiently small positive number  $\lambda$  results  $I_C = -\lambda (g, g) < 0$ , assuring a reduction in  $I_C$ . After making such a modification, the gradient can be recalculated and the process is repeated following a path of steepest descent until a minimum is reached. In order to avoid violation of the geometrical constraints, such as a minimum acceptable wing thickness, the

steps can be taken along the projection of the gradient into the allowable subspace of the control function. In this way, one can devise design procedures which must converge at least to a local minimum, and which might be accelerated by the use of more sophisticated descent methods. While there is a possibility of more than one local minimum, the cost function can be chosen to reduce the likelihood of difficulties caused by such a contingency, and in any case the method leads to an improvement over the initial design. The mathematical development resembles in many respects the method of calculating transonic potential flow proposed by Bristeau et al [17], who reformulated the solution of the flow equations as a least squares problem in control theory.

A historical summary of the design methods can be seen in Table 2.1 as they appear in [34]. More details about its method exist also in the above paper.

1945	LIGHTHILL	Conformal mapping, Incompressible inviscid flows
1965	NIEUWLAND	Hodograph, Power Series
1970	GARABEDIAN-KORN	Hodograph, Complex Characteristics
1974	BOERSTOEL	Hodograph
1974	TRENEN	Potential Flow, Dirichlet Boundary Conditions
1977	HENNE	3-D Potential Flow, Based on FLO22
1985	VOLPE-MELNIK	2-D Potential Flow, Based on FLO36
1979	GARABEDIAN-McFADDEN	Potential Flow, Newman Boundary Conditions, Iterated Mapping
1976	SOBIECZI	Fictitious Gas
1979	DRELA-GILES	2-D Euler Equations, Streamline Coordinates, Newton Iteration

**Table 2.1: Historical summary of the design methods**

### 2.2.3 Global vs. local methods

The fastest optimisation algorithms seek only a local solution, a point at which the objective function is smaller than at all other points in its vicinity. They do not always find the best of such minima, that is, the global solution. Global solutions are necessary (or at least highly desirable) in some applications, but they are usually difficult to identify

and even more difficult to locate. Only in the case of convexity, namely when the Hessian or the second derivative of the cost function has a specified sign, a local solution is also the global one [35].

Evolutionary algorithms and gradient-based methods are usually methodologies found in the literature to solve optimisation problems. These methods are different in the approach and solution of the optimisation problem as well the computational cost of each scheme. They can be distinguished by using the characterisation global methods or local methods. However, many successful global optimisation algorithms proceed by solving a sequence of local optimisation problems. A collection of research papers on global optimisation can be found in Floudas and Pardalos [36].

Global methods, such as those based on the genetic algorithm [37], are aiming to obtain the global optimum. These methods are most useful for cases in which multiple minima are present in the design space. It is widely known, however, that global methods incur large computational effort, where hundreds or even thousands of flow analyses may be needed before the global optimum can be found. In general, if the problem contains  $n$  design variables a genetic algorithm must perform  $n^2$  objective function evaluations to improve the original design.

Local methods use the information on the gradient of the objective for locating the optimum. Therefore, for cases with multiple minima, local methods are limited to produce only one of the minima (i.e., the local optimum), the actual value of which depends on the starting point of the optimisation process. Usually, a gradient-based approach is used because of its efficiency and low computational cost in yielding a minimum. Because of the reduced computational requirement, and since any local optimum represents an improvement over an existing design, local methods are very useful design tools. The method described in this thesis belongs to this category. Despite this efficiency, for optimisation problems involving a large number of design variables, it can be very difficult to obtain the descent direction and the step size that are needed to carry out the optimisation process. This is because sensitivity coefficients, that are information on how the flow variables are affected by each design variable, are needed.

The main complication of gradient based methods is the calculation of the required gradients. Recent developments in the gradient-based optimisation methodology suggest

that two main streams may be distinguished according to the way that these gradients are calculated: (i) the direct method and (ii) the variational method. This distinction is based on how the gradient is computed.

The gradient formulation, using direct methods [38-42], is done on a discrete level, which means that one must deal with the discrete form of the flow equations. The sensitivity derivatives on each grid point are calculated for each design parameter using finite differences. This method has the advantage that the sensitivities of the flow properties on the grid points can be determined. Once these become available, the gradient of a functional (cost function) can be computed easily using the chain rule. However, the computational effort strongly depends on the number of design variables. For each design variable, a sensitivity equation in the form of a (large proportional to grid size) linear system of equations must be solved. The computational requirement can therefore be prohibitive if a large design space is to be covered. The solution of a large linear system of equations to compute the flow variable gradient with respect to each design parameter is required. Thus for  $n$  design parameters  $n$  large linear systems have to be solved. This makes the method of order  $O(n+1)$  ( $n+1$  large linear system of equations have to be solved to evaluate the cost function and to obtain its gradient) and in general a detailed flow solution is relatively expensive for problems governed by the Euler and RANS equations. This cost can be reduced significantly if the  $LU$  decomposition method is used and the matrix is stored. However this alternative is presently only possible for 2-D problems.

The formulation of the gradient using the variational method can be done either on discrete level [43-45] or a continuous level [8, 46-49]. This method needs the values of the so-called adjoint variables as the solution of a set of adjoint equations. More details about this method exist in the model description Chapter 3 since this is the method used for the present studies. The numerical solution procedure for solving the flow equations can be adopted for solving the adjoint equations. The gradient is expressed in terms of the flow variables and the adjoint variables. The computational effort for obtaining the gradient is not determined by the number of design variables. Instead, it is determined by the number of adjoint equations that must be solved, which is equal to the number of functionals, either cost functions or constraints. Anticipating that the number of design variables is significantly larger than the number of functionals, which is true in most practical cases, the variational method has a significant advantage over the direct meth-

ods. Consequently, for problems involving many design parameters and few cost functions a better alternative is to employ a variational/inverse method (adjoint formulation). In this approach the effort to compute each cost function gradient requires one CFD solution for the usual variables and one for the adjoint variables, i.e. the cost is now only  $O(2)$  CFD solutions per design cycle.

#### 2.2.4 Gradient-based optimisation

Computational design has become an active area of research in the fluid dynamic community. Although simulation has been used in the design of fluid dynamic configurations in the past, only the optimisation cases that utilise sensitivity derivatives and gradient-based optimisation methods will be discussed in this section.

There is a huge number of methods, the so-called gradient-based, designed to optimise a smooth cost function  $I_C(D)$ , where  $D$  is the vector of design variables, given the ability to evaluate both  $I_C(D)$  and its vector gradient  $\nabla_D I_C$ . In unconstrained optimisation, the simplest method is to choose  $\nabla_D I_C$  as a search direction and use a simple one-dimensional optimisation method to find a local optimum and then repeat with the new gradient. A well-known class of algorithms for unconstrained minimisation of functions, whose gradient can be calculated, is the steepest descent method first proposed by Cauchy in 1847 [50]. The steepest descent method is particularly useful when the dimension of the problem is very large. On the other hand its main disadvantages are:

- (a) Each iteration is calculated independently of the others; that is, no information is stored and used that might accelerate convergence.
- (b) It is not generally a finite procedure for minimising a positive-definite quadratic form.
- (c) The rate of convergence depends strongly on the morphology of the objective function; if the ratio of the maximum to the minimum eigenvalue of the Hessian matrix of second derivatives  $\partial^2 I_C / \partial D_i \partial D_j$  of  $I_C$  at any local optimum is large, the steepest descent generates short zigzagging moves in a neighbourhood of the optimum [51]. More details about the steepest descent method exist in section 3.7 since this is the optimisation method used during this study.



More efficient methods approximate the Hessian matrix, effectively constructing a local quadratic approximation to  $I_C(D)$ . The approximation of the Hessian can be used to approximate for example the Newton step which accelerates the numerical solution of the optimisation problem. Arian [52] used the continuous adjoint method to solve a classical aerodynamic shape optimisation problem governed by the compressible Euler flow equations. The Hessian (second order derivative of the cost functional with respect to the shape variables) is approximated at the continuous level, as first introduced by the same author in [53]. In [54], sequential quadratic programming (SQP) method is used for the optimisation, in which the search direction is calculated by solving the quadratic sub problem where the Hessian is defined by a quasi-Newton approximation of an augmented Lagrangian merit function. The Lagrange multipliers in this merit function serve to scale the effect of any nonlinear constraints that the design may contain. Linear constraints are treated by solving the quadratic sub problem such that the projected search direction remains in feasible space. The above methods may be more efficient but the problem raised by using the adjoint technique to calculate gradients is that the approximation of the Hessian results in more steps in the optimisation procedure than those required for a direct sensitivity approach, like finite differences. Thus, the above optimisation methods are too costly to be applied to this study.

There are modifications to the standard unconstrained optimisation methods to handle constraints often through the use of Lagrange multipliers. However, when there is a large number of constraints, other approaches may be preferable. In particular, when the number of active constraints is equal to the number of active design parameters, the best approach may be to use a linear approximation to the objective function and constraints. The optimal solution to this linear programming problem can be found using the simplex method, and then the process is repeated using the new solution and the gradient information.

A more advanced and sophisticated method for gradient descent is the conjugate gradient method. This method is one of the most useful for solving large linear system and can also be adapted to solve nonlinear optimisation problems as well. It is actually being used as the solver of the linearised Navier-Stokes equations in this study. Since the method is more complicated than steepest descent and the present study mostly concerns about the gradient calculation technique, conjugate gradient method has not been

used for the optimisation. The conjugate gradient method is explained in full detail in [55].

There are also optimisation methods which are deterministic, but do not utilise gradient information. One of the most popular is, confusingly, also called the simplex method, or sometimes, the other simplex method to distinguish it from the linear programming problem [56].

As it has been mentioned in the previous section, the main complication and the most time consuming process in applying gradient-based optimisation in CFD is the calculation of the required gradients also called sensitivities. A detailed and concise overview of sensitivity analysis methods and aerodynamic design optimisation research may be found in Newman et al [57]. There are three well-known methods for addressing the sensitivity coefficient issue in gradient-based optimisation. These are the finite difference method, the direct-differentiation method, and the adjoint-variable method the way they are described by Tortorelli and Michaleris [58].

With the finite-difference method, a CFD simulation must be performed for each design variable to determine the sensitivity of that variable on the descent direction and step size. Since each CFD simulation can be very CPU and memory intensive, this approach is acceptable only if the number of design variables used to describe the geometry is small.

With the direct-differentiation method [39, 59-64], the chain-rule along with forcing functions are used to modify the governing equations so that sensitivity coefficients instead of the flow variables become the unknowns. This method is more accurate and more efficient than the finite-difference method. However, its cost is still proportional to the number of design variables. But, it has the advantage of having its cost being independent of the number of constraint equations. Thus, this method is ideal for problems with many constraints but few design variables.

Authors who dealt with direct differentiation and finite differences often use tools for automatic differentiation like *odyssee* which is a tool based on the ideas of Beux and Dervieux for the exact-gradient shape optimisation [44]. Pandya and Baysal [45] used ADI strategies for the shape optimisation in 3-D viscous applications. Other methods in

those categories are the alternative direction implicit method from the same authors [40] or the non linear projection from LeGresley [65] and the approximately factored increment strategy by Korivi et al [66]. Representative tools for Automatic Differentiation are Odyssee [67], ADJIFOR [68], TAMC [69].

With the adjoint-variable method, the need for sensitivity coefficients can be completely bypassed. This is accomplished by combining the objective function and the Navier-Stokes equations with a set of Lagrange multipliers or adjoint variables. The resulting equation is manipulated to eliminate all dependence on sensitivity coefficients, and this produces a set of adjoint-variable equations and the corresponding boundary conditions. To find the descent direction and step size, one only needs to obtain one CFD solution and one solution of the adjoint-variable equations. This implies that the cost of this method is independent of the number of design variables. Its cost, however, is still proportional to the number of constraint equations. Thus, the adjoint-variable method is ideal for problems with few or no constraints and a very large number of design variables.

The adjoint variable method can be formulated on a discrete level, as described by Forth et al [38] who used automatic differentiation for the extraction of the adjoint equations or a continuous level, like Jameson [6] who calculated the equations by hand. Comparing the two approaches, the discrete level is more accurate but less transportable than the continuous level. The two approaches are presented in more detail in section 2.3.1. The aforementioned tools for automatic differentiation apply again here in the discrete case. Since CFD shape optimisation can involve a very large number of design variables (e.g., thousands or more), the adjoint-variable method clearly has significant advantages over other methods. Detail description and discussion about the adjoint approach follows in the next section and the next chapters since this is the method used for gradient calculation in this study.

An alternative approach using the PDE sensitivity equation to develop algorithms for computing gradients is suggested by Borggard and Burns [70]. This approach produces consistent derivatives only in the case where asymptotically consistent schemes are combined with a trust region optimisation algorithm in a way that the resulting optimal design method converges. Also Hazra in [71] presents simultaneous pseudo-timestepping as an efficient method for aerodynamic shape optimisation. In this method, instead of solving the necessary optimality conditions by iterative techniques, pseudo-

time embedded nonstationary system is integrated in time until a steady state is reached. The main advantages of this method are that it requires no additional globalisation techniques and that a preconditioner can be used for convergence acceleration which stems from the reduced SQP method. The important issue of this method is the trade-off between the accuracy of the forward and adjoint solver and its impact on the computational cost to approach an optimum solution.

Several application of gradient-based methods used for CFD optimisation exist in the literature. Martin and Dulikravitz in [72] demonstrated the computational advantages of using implicit sensitivity with the boundary element method (BEM) method for the purpose of providing internal cooling systems designers the ability to optimise the internal cooling configuration, geometry and heat transfer enhancements for greater cooling efficiency and more durable turbine airfoils. Results show a three-to-one improvement in the optimisation convergence rate and greater gradient accuracy were obtained for the two-dimensional thermal optimisation problems. Balagangadhar [73] augmented the analysis capabilities of a CFD code along with design sensitivity analysis (DSA) and a numerical optimisation scheme. The sensitivities are calculated using analytical differentiation methods and the whole algorithm is applied to the optimisation of heat exchanger fin and HVAC duct systems. A method based on flow simulations and gradient-based optimisation techniques is presented by Kameron et al in [74] introducing the components of the fully automated optimisation loop and their interactions. This approach uses the fully parameterised blade geometry as variables for the optimisation problem. Physical parameters such as stagger angle, stacking line, and chord length are part of the model. The gradients required for the optimisation algorithm are computed by numerically solving the sensitivity equations and validated by means of finite differences. Shankaran et al [75] present a sail shape optimisation method, which combines the commercial CFD package FLUENT with gradient-based cost function minimisation. Results are presented for the optimisation of sheeting angles for the rig of a three masted clipper yacht. They investigated two cost functions, both characteristic of a sail's aerodynamic performance: the reciprocals of the driving force coefficient and the ratio of driving to heeling force coefficients. Comparing results for upwind and close reaching apparent wind angles shows that the latter leads to well trimmed sails, whereas the former causes the sails to be over trimmed.

An optimisation procedure for helicopter rotor aerodynamic performance is presented in [76]. This optimisation procedure is centred on the hybrid numerical optimiser CONMIN. The optimisation methods can be divided in two main types: the deterministic methods that reach the nearest local optimum and the non-deterministic methods that are conceived to reach the global optimum. The gradient-based method minimises a functional under constraints using feasible directions while the gradients are calculated with finite differences. The optimiser has been coupled to a 3-D Navier–Stokes CFD solver *elsA*, and applied to helicopter rotor optimisation in hover. Svenningsen et al [77] reduced the computational time of the optimisation using a quasi-analytical method rather than the time-intensive approximation by finite differences. The optimisation tool was applied on a two-dimensional laminar diffuser in order to obtain maximum pressure recovery by contouring the divergent wall section. This investigation yields a diffuser performance improved by about 5% when compared with a straight-walled geometry.

A general and efficient method for gradient based shape optimisation of fluid flow and FSI problems is presented in [78]. Expressions for design sensitivity analysis (DSA) are derived using the direct differentiation approach and the use of an inexact Jacobian matrix in the analysis leads to an iterative but very efficient scheme for DSA. Li et al in [79] presented a modified profile optimisation method using a smoother shape modification strategy for airfoil shape optimisation in a preliminary design environment. The method is formulated to overcome two technical difficulties frequently encountered when conducting multipoint airfoil optimisation within a high-resolution design space: the generation of undesirable optimal airfoil shapes due to high frequency components in the parametric geometry model and significant degradation in the off-design performance. The novel ideas used in the proposed method, such as the smoothest shape modification and modified profile optimisation strategies, are applicable to minimising aircraft drag at multiple flight conditions.

A two-dimensional aircraft high-lift system design and optimisation method is presented in [80]. The method uses a gradient based local optimiser based on modified feasible directions which uses Finite Differences for the calculation of the gradients. The representation of airfoils by general shape functions as well as element positioning was considered. Results demonstrated the appropriateness of the approach for high lift system design and optimisation.

### 2.2.5 Optimisation methods based on evolutionary computation algorithms

Although the optimisation method developed in this research belongs to the deterministic methods (gradient-based), a summary of work in stochastic methods (evolutionary algorithms) is presented here for completeness.

Stochastic methods, like genetic algorithms and related evolutionary algorithms introduce a random element in the evolution of the design. They evaluate whether or not the new designs are better and feasible, allowing some infeasible and poorer solutions in the short term but aiming for feasibility and optimality in the long term.

Evolutionary computation (EC) algorithms are inspired by nature's capability to evolve living beings well adapted to their environment that cooperate or compete with other members of the population. EC algorithms can be characterised as computational models of the evolutionary process that take inspiration from the natural genetic variety and natural selection. In some optimisation problems, the model cannot be fully specified because it depends on quantities that are unknown at the time of formulation. However, modellers can predict or estimate the unknown quantities with some degree of confidence. This is succeeded by introducing possible scenarios for the values of these quantities or even by assigning a probability to each scenario. Stochastic optimisation algorithms use these quantifications of uncertainty to produce solutions that optimise the expected performance of the model whereas in deterministic optimisation the model needs to be fully specified.

In every iteration of the EC algorithm a number of operators are applied to the individuals of the current population to generate the individuals of the population of the next generation (iteration). Usually, EC algorithms use operators called recombination or crossover to recombine two or more individuals to produce new ones. They also use mutation or modification operators which cause a self-adaptation of individuals. The driving force in EC algorithms is the *selection* of individuals based on their fitness (which can be based on the objective function or some other kind of quality measure). Individuals with a higher fitness have a higher probability to be chosen as members of the population of the next iteration (or as parents for the generation of new individuals). This corresponds to the principle of survival of the fittest in natural evolution.

There has been a variety of different EC algorithms proposed over the years. Basically, they fall into three different categories which have been developed independently of

each other. These are evolutionary programming [81], evolution strategies [48], [82], and genetic algorithms [83-85]. Variations and combinations of these methods have appeared in the literature like the case of the combination of genetic algorithms with neural networks by Giannakoglou et al [37]. The neural networks are trained within the generations of the GAs to replace the RANS calculations. Results of this method show significant decrease of the overall computational time of the optimisation procedure. Also the flow field is accurately approximated by the neural network bases functions. A hybrid aerodynamic shape optimisation tool couples a stochastic genetic algorithm and a deterministic/gradient-based BFGS hill climbing method for industrial flows is presented in [86]. For each evaluation required by the optimiser, the Navier–Stokes equations with the  $k$ – $\epsilon$  turbulence model are solved with a commercial CFD code on an unstructured mesh. The method was applied to minimise the drag coefficient by optimising the rear of a simplified car shape.

In [87] the solution of two multi-objective and multi-disciplinary shape optimisation problems related to industrial CFD using genetic algorithms is considered. The first one is a single objective optimisation problem, where the geometry of a flow divider of a paper machine head box is designed subject to prescribed goals and restrictions. The second problem is a two-dimensional airfoil design problem, where the objectives are to minimise the drag while the lift is kept larger than a given value.

In order to improve efficiency of a system with three-dimensional flow characteristics, Han and Maeng [88] presented a new method that overcomes the computational difficulties associated with three-dimensional effects by using two-dimensional CFD and a neural network. Results show the efficiency of the method. An optimisation environment for multidisciplinary turbomachinery blade design based on evolutionary strategies which simplifies the use of various optimisation and evaluation methods is presented in [89]. The algorithm consists of a blade parameterisation scheme in order to perform particular blade modifications with a limited number of optimisation parameters, a parallel CFD solver and a pre-evaluation tool which extracts necessary data and processes objectives and constraints to be assessed in an evaluator indicating the design quality. The developed optimisation environment enables sensitivity analyses as well as optimisation runs.

Over the years there have been quite a few overviews and surveys about EC methods. Among these are the ones by Bäck [90], by Fogel [91], by Spears et al [92] and by Michalewicz and Michalewicz [93]. In Calegari et al [94] taxonomy of EC algorithm is proposed.

## 2.3 ADJOINT-BASED OPTIMISATION

As it has already been discussed in the first chapter, the problem that we are dealing with in this study is the cavitation inception control of diesel nozzle injectors. A gradient-based method has been chosen for the optimisation due to its accuracy and fast convergence. In nozzle re-design cases the required parameters describing the geometry and affecting cavitation may be numerous. A detail discussion about these parameters follows in section 3.5. The computational cost when using a direct method to obtain the required gradient information is proportional to this large number of design parameters. This fact disabled methods as the most commonly used finite differences and motivated the development of adjoint methods for the calculation of the gradient.

The adjoint method has a number of advantages relative to other gradient-based methods, for example finite differences. Apart from its fast convergence, it provides the gradients of the cost function in a way that the computational effort required for this calculation is independent of the number of the design variables [95]. Of course the possibility of getting trapped into local minima exists as in every gradient based method. In the adjoint method the governing flow equations are treated as constraints by adding them to the cost function through Lagrange multipliers providing the augmented form of the cost functional. By taking the variational of the augmented cost function and the consequent limitation of the flow field variations, the variation of the adjoint variables as well as the sensitivity derivatives of the cost function with respect to the design variables are obtained. The gradient of the cost function at each location of the design space is dependent on the flow field and the costate variables distribution along the wall to be designed. So, the efficient solution of the flow and the adjoint equations may lead to the calculation of the exact values of the sensitivity derivatives. Details about the applications of the adjoint method exist in the following subsections.



### 2.3.1 The discrete and the continuous approach

The followers of the adjoint method are divided into two major groups according to their preference in the continuous or discrete adjoint method. By definition, when the "continuous" approach is used, one formulates the adjoint PDE and then approximates it using an appropriate discretisation; this discretisation might be entirely different to that used for the nonlinear PDE. Whilst, using the "fully-discrete" approach to approximate the adjoint equations, one starts with the discretisation of the original nonlinear PDE, and then linearises it and forms its transpose equation [96]. The resulting discrete adjoint equations represent a possible discretisation of the adjoint partial differential equation. If these equations could be solved exactly they could provide the exact gradient of the cost function which results from the discretisation of the flow equations, which is itself, however, inexact; they would be identical to finite difference sensitivities obtained from the flow calculation using infinitesimal step size on a computer with infinite machine precision. On the other hand, any consistent discretisation of the adjoint partial differential equation will converge to the exact gradient as the mesh is refined. One could even follow an intermediate path, linearising the original equations, discretising them and then taking the transpose equation. In principle, considering that each of the steps is performed correctly, and all of the solutions are sufficiently smooth, then for infinite grid resolution all three approaches should be consistent and converge to the correct analytic value for the gradient of the objective function. However, there are conceptual differences between the different approaches, and for finite resolution grids there are also differences in the computed results.

The major advocate of the continuous approach is Jameson [97] while of the discrete is Giles [96]. Jameson along with Nadarajah investigated the differences of the two approaches. In [98] they present a complete formulation for the continuous and discrete adjoint approach to automatic aerodynamic design using the Euler equations. The gradients from each method are compared to finite difference gradients and the following conclusions are drawn:

1. The continuous boundary condition appears as an update to the costate values below the wall, and the discrete boundary condition appears as a source term in the cell above the wall. As the mesh width is reduced, one recovers the continuous adjoint boundary condition from the discrete adjoint boundary condition.
2. Discrete adjoint gradients have better agreement than continuous adjoint gradients with finite difference gradients as expected, but the difference is generally small.

3. As the mesh size increases, both the continuous adjoint gradient and the discrete adjoint gradient approach the finite difference gradient, whilst the difference between them, obviously, decreases.
5. The computational cost of deriving the discrete adjoint as well as the memory required is much greater.
6. The overall convergence of the objective function is not significantly affected when the discrete adjoint gradient is used instead of the continuous adjoint gradient.

Jameson and Vassberg [9] stated the benefits gained from developing the theory for the partial differential equations of the flow. First, the true optimum shape belongs to an infinitely dimensional space of design parameters, and the theory provides an indication of how such a solution could be approached if sufficient computational resources were available. Second, it provides insight into the nature of the adjoint equations, and the connection between the formulation of the cost function and the boundary conditions needed to assure a well-posed problem. Third, in certain circumstances the discrete solution may lose the property of continuous dependence of the design parameters. It may, for example, contain non-differentiable flux limiters. Also, if adaptive mesh refinement is used, there will be a discontinuous change in the solution whenever a mesh point is added or deleted. Finally, discretisation methods for differential equations provide guideline for the design of iterative solutions for the adjoint equation; either for the case where the adjoint equation is separately discretised or for the case where the discrete adjoint equations are derived directly from the discrete flow equations.

Consequently, according to Jameson and his followers, there is no particular benefit in using the discrete adjoint method, which requires greater computational cost. On the other hand, Giles states that the continuous adjoint approach yields a discrete approximation to the gradient of the analytic objective function with respect to each of the design variables. This will not be exactly equal to the gradient of the discrete approximation to the objective function. Therefore, there is a slight inconsistency between the discrete objective function and the computed gradient. As a result, the optimisation process will fail to converge further once the solution is near a local minimum.

There are several pragmatic issues that led Giles to prefer the discrete approach. One is the fact that the "fully-discrete" method is completely prescriptive; there is a straightforward process by which one generates the adjoint code, and this can be significantly aided by the use of Automatic Differentiation (AD) [99] techniques to automatically

generate key pieces of the adjoint code. Another is that it is possible to solve the discrete adjoint equations using an adjoint version of the highly-optimised iterative methods developed for the nonlinear equations. This is the subject of papers [100, 101], with [96] also discussing the complexities that arise with the imposition of strong boundary conditions.

The advantages of the fully-discrete approach could be briefly summarised as:

- The exact gradient of the discrete objective function is obtained. This ensures that the optimisation process can converge fully. It also provides a convenient check on the correctness of the programming implementation; with the continuous approach one does not know whether a slight disagreement is a consequence of the inexact gradient or a possible programming error.
- Creation of the adjoint programme is conceptually straightforward. In the future this should enable the almost automatic creation of adjoint programmes using AD software. This benefit includes the iterative solution process since the transposed matrix has the same eigenvalues as the original linear matrix and so the same iterative solution method is guaranteed to converge.

On the other hand, the advantages of the continuous approach are:

- The physical significance of the adjoint variables and the role of adjoint boundary conditions are much clearer. Only by constructing the adjoint flow equations can one develop a good understanding of the nature of adjoint solutions, such as the continuity at shocks, etc.
- The adjoint programme is simpler and requires less memory. Because one is free to discretise the adjoint PDE in any consistent way, the adjoint code can be much simpler. However, even when following a continuous approach, it is advantageous to use the same discretisation for the continuous adjoint equations as for the flow equations. It is also generally the case that continuous adjoint solvers require less memory than the fully-discrete codes.

Recently, another comparison of the two approaches is given by Duivesteijn et al [102]. They also dealt with the problem of boundary condition treatments applied to the quasi-1D Euler equations. The effect of strong and weak boundary conditions and the effect of flux evaluators on the numerical adjoint solution near the boundaries were discussed.

There are many applications of the adjoint method over the years in shape optimisation using either the discrete or the continuous method. In this subsection some of these are summarised giving emphasis on the approach used.

A continuous adjoint method is presented in [103]. The optimisation algorithm sequentially builds an approximate optimisation sub-problem, based on the value of the functions and their gradient at the current design point solved with a dual method. Various approximations are investigated, including the method of moving asymptotes (MMA) and diagonal quadratic methods. Monge and Tobio in [104] used Jameson's approach to develop the adjoint equations for the incompressible two-dimensional flow with a given velocity distribution as objective, and the aim of extending it in short term to the use of friction drag. Their main purpose was to compare the results provided by control theory with those of finite differences in terms of gradient evaluation and computing time. The drag minimisation of a body which initial shape was a circular cylinder using control theory for the incompressible Navier-Stokes equations is investigated by Ochiai and Kawahara in [105].

In [106] Li and Padula presented a robust optimisation methodology based in the discrete adjoint configuration. They concluded that the importance of their method increases when the parameterisation has strong non-linear effect to the cost function. Another author who is in favour of the discrete approach is Muller, who in [107] reviews the implementation of Automatic Differentiation (AD) [99] on Fortran CFD codes and gives details of how small rearrangements can be used to produce competitive tangent and adjoint code using AD. Forth [38] uses the *Odyssee* AD tool to obtain and validate a discrete forward sensitivity version of the BAE SYSTEMS/AIRBUS UK unstructured mesh CFD code Flite3-D. A very useful paper for the discrete approach followers is the paper of Giering et al [69] which gives information about the construction of discrete adjoint codes and tools for automatic differentiation although the applications are completely different to those of the present study.

A discrete approach which gives the ability to solve simultaneously for the flow and adjoint equations has been developed by Kuruvila et al [108]. The basic idea of this *one-shot* method is to change the shape of the airfoil profile in a hierarchical manner such that smooth changes are made separately from high frequency changes. Because each of these changes involves a different scale, the governing equations of the flow can

be solved efficiently on grids of appropriate resolution. Thus, the flow field due to smooth changes in the airfoil is solved on coarse grids, and the flow field to increasingly fine grids. The resulting optimisation procedure breaks into a sequence of sub-optimisation problems solved on the appropriate grid. The *one-shot* method is implemented in a multigrid algorithm. Results showed that the method brings the cost of the overall optimisation problem to the cost of the flow solution. But the method has not been used to solve practical problems apart from simple airfoil design.

Some authors used both approaches. Anderson, who was one of the first who extracted the continuous adjoint equations for inviscid flows independent from the type of the mesh [7], moved to the discrete approach when along with Nielsen [109] they constructed a parallel framework for performing aerodynamic design optimisations on unstructured meshes. Only the inviscid terms were treated in order to develop a basic foundation for a multiprocessor design methodology. A parallel version of the adjoint solver was developed using a library of MPI-based linear and nonlinear solvers known as PETSc, while a shared-memory approach is taken for the mesh movement and gradient evaluation codes. Parallel efficiencies were demonstrated and the linearisation of the residual was shown to remain valid.

Recently, a team in DLR [110] gave an overview of the capabilities of the discrete adjoint to perform aerodynamic shape optimisation in viscous flow. They developed a discrete adjoint method implemented on their in-house Navier-Stokes unstructured finite-volume solver, the DLR-TAU-code. The method consisted of the explicit construction of the exact Jacobian of the spatial discretisation with respect to the unknown variables allowing the adjoint equations to be formulated and solved. A wide range of the spatial discretisations available in TAU have been differentiated, including the Spallart-Almaras-Edwards one-equation, and the Wilcox  $k - \omega$  two-equation turbulence models. The strategy developed was validated only on 2D cases and compared with the approximate gradients obtained by finite-differences. Then the accuracy and efficiency of the approach were demonstrated for transonic airfoil design by considering geometric as well aerodynamic constraints, single- as well as multi-point design. Finally, the flap design of a multi-element airfoil in take-off configuration confirmed the capability of the discrete adjoint to solve a wide range of aerodynamic problems.

A methodology to mitigate the development cycle typically associated with constructing a discrete adjoint solver was presented in [111]. The approach was based on a complex-

variable formulation that enables straightforward differentiation of complicated real-valued functions. An automated scripting process was used to create the complex-variable form of the set of discrete equations. The accuracy of the implementation was verified through comparisons with a previously developed discrete direct method.

Another discrete-adjoint formulation for the 3-D Euler equations discretised on a Cartesian mesh with embedded boundaries was presented in [112]. The solution algorithm for the adjoint and flow-sensitivity equations leveraged the Runge–Kutta time-marching scheme in conjunction with the parallel multigrid method of the flow solver. The matrix-vector products associated with the linearisation of the flow equations were computed on-the-fly, thereby minimizing the memory requirements of the algorithm at a computational cost roughly equivalent to a flow solution. A wing-body geometry at transonic flow conditions case and an entry vehicle at supersonic flow conditions case verified the accuracy of the linearisation and demonstrated the efficiency and robustness of the adjoint algorithm for complex-geometry problems.

Although we have aimed to be objective in our assessment of the relative advantages of the two approaches, it should be noted that the author is an advocate of the continuous approach. An advocate of the discrete approach may place a different emphasis on the above observations and hence reach a different conclusion. Certainly, both methods perform well in practice, and it remains to be seen whether either approach demonstrates compelling advantages over the other in terms of design performance. After analysing the pros and cons of both methods the author would like to emphasise on the saying of Giles [113]: “Ultimately, the final choice may always remain, to some extent, a matter of personal taste”.

### **2.3.2 Properties of the adjoint solutions**

The scope of the present research is to apply the adjoint method in order to optimise Diesel nozzle geometries according to specified flow characteristics. As a result, the author has not dealt with the problem of the properties of the adjoint solutions. For that reason this section comprises a detailed discussion of the existing literature in this area about the underlying theory, formulation, approximation and solution of adjoint equations.

The first who dealt with the mathematical aspect behind the adjoint equations was Giles, who, in 1997 along with Pierce considered the proper formulation of adjoint boundary conditions for the Euler and Navier-Stokes equations [114]. They proved that there is a limited set of objective functions for which the standard formulation of the adjoint problem is well posed. In the same paper they also identified some key properties of the solutions to the adjoint quasi-1-D and 2-D Euler equations. The restrictions on the permissible choices of cost functions show that at solid walls these cost functions must be a function of the linearised pressure for inviscid flows or a function of the normal and tangential forces for viscous flows to lead to a well posed problem. Also Jameson et al [115] mentioned that the boundary conditions satisfied by the flow equations restrict the form of the adjoint boundary condition. Consequently, the boundary contribution to the cost function cannot be specified arbitrarily. Instead, it must be chosen from the class of functions which allow cancellation of all boundary terms resulting from the adjoint analysis.

Arian and Salas [116] contradicted this and proposed a method so those “inadmissible” cost functions can actually be “admissible”. They showed that for problems with inadmissible cost functions, additional terms should be included in the Lagrangian functional when deriving the adjoint equations. These terms can be obtained from the restriction of the interior PDE to the control surface. They demonstrated the explicit derivation of the adjoint equations for “inadmissible” cost functionals for the potential, Euler, and Navier-Stokes equations. Anderson [7] and Soemarwoto [117] also offered a solution by adding terms to the inadmissible cost functions which are actually zero but allow the cancellation of all the boundary terms. Their method is used in the present studies where the cost functions are considered inadmissible and otherwise could lead to ill-posed problems.

The problem of the adjoint solutions in cases where there are discontinuities in the flow was the basic issue at the beginning of Giles involvement in adjoint optimisation [114]. In the first half of that paper, he derived the adjoint equations for inviscid and viscous compressible flow with the emphasis being on the correct formulation of the adjoint boundary conditions. It is shown that the boundary conditions for the adjoint problem can be simplified through the use of a linearised perturbation to generalised coordinates. In the second half of that paper, the Green’s functions were constructed for the quasi-1-D and 2-D Euler equations. These were used to show that the adjoint variables have a

logarithmic singularity at the sonic line in the quasi-1-D case, and a weak inverse square root singularity at the upstream stagnation streamline in the 2-D case. However, they are continuous at shock locations in both cases. These conclusions led to the derivation of a closed form solution for the adjoint quasi-1-D Euler equations with and without shocks [118, 119].

The properties of adjoint solutions were examined for the quasi-1-D Euler equations, in more detail in [118]. For shocked flow, the derivation of the adjoint problem reveals that the adjoint variables are continuous with zero gradients at the shock, contradicting previous results of Iolo and Salas [120], and that an internal adjoint boundary condition is required at the shock. The analytic adjoint solution was then derived for isentropic and shocked transonic flow revealing a logarithmic singularity at the sonic point. Numerical experiments with both the discrete and continuous adjoint formulations suggested that the adjoint solution behaves correctly at the shock without explicit enforcement of the internal boundary condition. For the continuous approach, in the absence of explicit enforcement of the correct adjoint boundary conditions at the shock, the correct asymptotic behaviour can be explained as the effect of numerical smoothing, given that the correct analytic solution was the only smooth solution at the shock. In the same paper an adjoint approach to a posteriori error analysis was then demonstrated and the implications for developing an optimal adaptive algorithm were discussed.

In [119] a Green's function approach was used to derive the analytic adjoint solutions corresponding to supersonic, subsonic, isentropic and shocked transonic flows in a converging-diverging duct of arbitrary shape. This analysis revealed a logarithmic singularity at the sonic throat and confirms the expected properties at the shock. It remains an open question whether the continuous or the discrete approach is better when there are nonlinear discontinuities such as shocks. For quasi-1-D Euler calculations, for which they derived the analytic solution of the adjoint equations [118], both approaches give numerical results which converge to the analytic solution. For the discrete approach, this follows because the integrated pressure can be proved to be predicted with second-order accuracy [121]. The linearised discretisation should therefore yield perturbations to the integral of pressure that are at least first-order accurate.

In 2-D and 3-D there is no proof of second-order accuracy for quantities such as lift and drag, and there is a discontinuity in the gradient of the adjoint variables at the location of the shock. Therefore it remains an open question as to whether either approach, dis-



crete or continuous, will give a consistent approximation to the gradient of the objective function in the limit of infinite grid resolution. However, practical results for applications with weak shocks suggest that any inconsistency must be small. In practice, researchers using the continuous adjoint approach do not enforce these special boundary conditions for the discontinuities, and their results indicate no difficulties as a consequence.

As we mentioned in the previous subsection, Giles is an advocate of the discrete method so in his research he often dealt with problems faced when applying this approach. In [101] he showed that the naive application of the adjoint method to the iterative solution of a linear system of equations may produce an algorithm which does not correspond to the iterative solution of the corresponding adjoint system of equations. An algebraic manipulation that can transform the algorithm so that the adjoint variables converge to the solution is proposed. Using this manipulation also a continuation in the adjoint solution procedure may be applied. Giles also wanted to explain the key ideas of optimal design using adjoints. So he introduced the mathematics at the level of linear algebra, where things are relatively simpler, before progressing to partial differential equations, as an easy way to explain the adjoint approach to design and error analysis [113].

Giles latest research interest is in what happens when the underlying nonlinear solution, in the discrete approach, has a shock, and the conditions under which the computed adjoint solution will approach the analytic solution as one refines the computational grid. His first paper on this topic [122] deals with numerical results indicating the necessity of grid resolution improvement. The numerical smoothing varies in a way which increases the number of points across the shock, while at the same time the overall width of the shock decreases, to get a convergent adjoint discretisation. Another fundamental conclusion for any numerical discretisation with a fixed number of points across the shock is that, it is easy to fall in the trap of constructing an objective function for which the numerical adjoint solution will not converge.

Jameson also dealt with the properties of the adjoint solutions. Along with Kim [123] they investigated the accuracy of the resulting, from the adjoint analysis, derivative information by direct comparison with finite-difference gradients and by mesh and parameter refinement studies. In the process, shortcomings of the finite difference method for the calculation of derivative information are pointed out and discussed. The advan-

tages of the use of an adjoint method become apparent because of the strict requirements that the finite difference method imposes on the level of flow solver convergence and the sensitivity of the value of the gradients with respect to the choice of step size. From the analysis it was concluded that the gradient information obtained using the adjoint method is much less dependent on the level of convergence of the flow solver and it is insensitive to the step size chosen in the deformation of the aerodynamic configuration. Also the adjoint method requires only modest levels of convergence of the adjoint solver, thus reducing even further the computational cost of this procedure. In another case along with Vassberg [124] they used the simple Brachistochrone problem which has an analytic solution in order to further investigate the accuracy, performance and robustness of several optimisation techniques. Results showed that, at least in the case of the Brachistochrone, the continuous gradient yields a slightly more accurate solution than the discrete gradient.

Protas and Bewley [125] summarised the taxonomy of regularisation opportunities available in the adjoint analysis of multi-scale fluid systems. They developed a uniform framework for understanding the well-posedness and regularisation issues of the adjoint solutions. These are given in the following four ways:

1. by adding an artificial term to the discretised state equation itself.
2. by the definition of the cost functional.
3. by the form of the duality pairing.
4. by the definition of the inner product used to extract the functional gradient.

### **2.3.3 Applications of the adjoint – based optimisation**

Over the past two decades many investigators have employed the adjoint-variable method in CFD shape optimisation either in the discrete or the continuous mode and benefited by the advantages vs. other most traditional methods. Some of the features of the most representative applications of the adjoint method in shape optimisation are summarised in this subsection.

There is a long history of the use of adjoint equations in optimal control theory [33]. In fluid dynamics, the first use of adjoint equations for design was by Pironneau [4, 5], who as early as the 1970's, suggested an optimal shape design algorithm for the first time and derived the optimality condition for a body with minimum drag in low-Reynolds-number laminar flow. However, within the field of aeronautical computa-

tional fluid dynamics, which is the field with most applications, the use of adjoint equations has been pioneered by Jameson [6], who in the late 1980s used his knowledge of optimal control theory to develop what he called optimal design methods. During the past few years he applied the adjoint variable method for optimal shape design of airfoils, wings and complete aircrafts using the Euler [6, 126, 127] or the Navier-Stokes equations [115, 128]. During his research he collaborated with a lot of other investigators. With Reuther et al [46, 54, 129] they did some publications for complex wing geometries, wing-body and whole aircraft geometries where they enforced geometrical constraints. They worked mostly using the Euler equations and for the acceleration of the algorithm multigrid and parallelisation using MPI have been implemented in the solver. With Kim et al they did some theoretical work [123, 130] which has been referenced in the previous section, concerning the accuracy of the adjoint gradients and the choice of optimisation parameters. They also worked on the optimisation of high-lift configurations [131, 132] using the adjoint method for the Navier-Stokes equations with the Spalart-Allmaras turbulence model. Optimisation was performed in the sense of drag coefficient reduction with constant lift or drag to lift ratio. Also validation of the model was achieved by the inverse design of the airfoil using a prescribed pressure distribution. In his latest work with Nadarajah [133, 134] he used the discrete adjoint approach to achieve sonic boom reduction in a supersonic aircraft. The challenge of this approach is the calculation of remote sensitivities; the cost function and thus the gradients are defined on a different boundary from the boundary of the shape to be modified. A similar challenge has been faced in the present study in the case of  $C_d$  maximisation. Details concerning the handling of this problem exist in the following chapter.

Anderson in 1997 extracted the continuous adjoint equations for inviscid flows independent from the type of the mesh [7]. He also included the analysis for the incompressible laminar flows in the same paper. In [43] he included in the adjoint analysis turbulence modelling as well. Ta'asan et al were also from the first researchers that used the adjoint formulation and control theory for shape optimisation in inviscid flows [53, 135, 136]. Along with Iolo they extended their theory in more complex applications [137, 138] and also they tried to extend the method for transonic flows [120]. Giles was the one who first dealt with the problem of boundary conditions for the adjoint equations and proved the duality of the adjoint and flow equations [114]. He studied the behaviour of the analytic solution of the adjoint equations in quasi 1-D [119] and 2-D [118, 122] and he reached to the conclusion that contrary to [120] the adjoint variables

don't appear to have discontinuities in the transonic cases. He mostly dealt with the Euler equations and the discrete approach. In the developed discrete adjoint he reduced the computational cost by algebraic manipulation that allow continuation in the optimisation iterations [101] and he formulated a preconditioning Runge-Kutta with multigrid in a way that equal numbers of iterations of the direct and the adjoint solver will result in the same value for the linear functional being sought [100].

In 1997 he gave a lecture in VKI [139] in design using adjoint for aircraft and turbomachines where he mentioned the advantage of greatly reduced computational cost when dealing with large number of design parameters. In [56] he identified some of the complexities inherent in the design of aeroengines and aircraft. A range of different numerical approaches to design were outlined and their strengths and weaknesses were compared. Other related topics such as distributed computing, risk management and strategic research planning were also discussed. Along with Duta et al [140] they developed a discrete adjoint code for the Euler and the Navier-Stokes equations where they enforced strong boundary conditions and associated conditions for lift and drag, using a particular form of Runge-Kutta. In this first harmonic adjoint algorithm the results were exactly equivalent with the linear perturbation code for each individual iteration. This guarantees the same rate of iterative convergence which is very useful for validation and debugging of the adjoint code. In the next paper with Campobasso et al [141] they validated the above results by checking the adjoint's exact equivalence to the linear solver in steady and unsteady cases for the unsteady design of turbomachinery blades. The same authors [142] applied for the first time the adjoint method to the linearised analysis of periodic unsteady flows by involving tailoring of incoming wakes to reduce the level of forced response blade vibrations. The linear flow analysis of turbomachinery aeroelasticity views the unsteady flow as the sum of a background nonlinear flow field and a linear harmonic perturbation. The background state is usually determined by solving the nonlinear steady flow equations. The flow solution representing the amplitude and phase of the unsteady perturbation is instead given by the solution of a large complex linear system which results from the linearisation of the time-dependent nonlinear equations about the background state. In this first approach to solve unsteady problems they pointed at the physical origin of numerical instabilities appearing due to the preconditioned multigrid iteration of the parallel flow solver. In a recent paper [143] they achieved the code stabilisation by using the real and complex GMRES and RPM algorithms to stabilise the existing preconditioned multigrid iteration. This approach consid-

ered an equivalent augmented real form of the original complex system of equations. Results showed that the complex solvers not only stabilised the code, but also led to a substantial enhancement of the computational performance with respect to their real counterparts.

After Huan and Modi [144] who performed drag minimisation in laminar flows, Soemarwoto [117, 145] extended the adjoint method in viscous flows. Until then, most investigators used a simplified form of the Navier-Stokes equations such as the potential or Euler equations. With simpler governing equations for the flow field, the adjoint variable formulation is made simpler. Soemarwoto [145] demonstrated the feasibility of the approach in dealing with inverse problems and constrained drag-reduction problems, where the compressible viscous flow model based on the RANS equations was used for the flow calculations. An analytical expression of the adjoint equations was formulated based on the continuous form of the aerodynamic functional and the RANS equations. However, considerations from the physics of the boundary layer aren't taken into account for obtaining an approximation of the gradient, despite the success in obtaining true viscous adjoint solutions. Although the approximation can lead to useful results, as shown in Ref. [119], it is desirable to have a gradient expression which is derived consistently using the RANS equations. This work was extended by the same author in [117] where he computed optimal shapes of airfoils using the compressible Navier-Stokes equations.

In their first publication Soto and Löhner [146] used the continuous adjoint approach to extract incomplete gradients for the incompressible Euler equations. Several examples indicate the present scheme yields proper results without incurring the cost of a complete adjoint solution. The method was extended [147] for the Navier-Stokes equations where the continuous and the discrete approaches were presented. Also an innovative and very fast volume mesh movement algorithm was developed, which allows including the interior point contributions to the gradient. A very fast pseudo-shell approach that produces smooth singularity-free shapes was presented [148] to parameterise surfaces in CFD optimisation problems. The user has to generate only the original surface mesh and a few design variables. The rest of the design parameters and their respective deformation modes can be generated automatically by the method. In [149] they included turbulence modelling but only for the discrete approach. For the mesh movement, a deformation mode is used with a quasi-incompressible elastic scheme for the elements outside

of the boundary layer region. Such hybrid technique was tested in three problems containing highly stretched elements with excellent results. A summary of their work for complete continuous adjoint spanning compressible and incompressible viscous and inviscid exists in [150]. Recently [151], they extracted general continuous expressions using only values of adjoint and flow variables at the surface where objective function is defined. In that work they noticed that an error can occur from the implementation of different types of incomplete gradients.

In his latest work Kim, et al [130] has investigated the feasibility of improving the existing adjoint method by finding an optimal combination of flow analysis and design input parameters. They enhanced the existing adjoint-based design method by determining the control parameters of CFD analysis and design process by the use of the non-linear gradient-based optimiser package, SNOPT. SNOPT is a gradient-based package that uses a sequential quadratic programming (SQP) algorithm that obtains search directions from a sequence of quadratic programming sub-problems. Each QP sub-problem minimises a quadratic model of a certain Lagrangian function subject to a linearisation of the constraints. The numerical results show the adjoint design method can be improved in shape design speed, performance, and stability by integrating the method with a parameter optimisation tool such as, but not limited to, SNOPT.

Recently, Alonso and LeGresley [152] have implemented POD as a method of reducing the coupling bandwidth between disciplines in a decomposition method called Bi-Level Integrated System Synthesis (BLISS). By using POD the normally high bandwidth interaction between some disciplines, such as aerodynamics and structures where all of the surface pressures and structural displacements need to be exchanged, is replaced by a one time exchange of modes and a per iteration exchange of values that scale the modes.

Hazra and Schultz [153] presented a numerical method for constrained aerodynamic shape optimisation problems. It was based on simultaneous pseudo-timestepping in which stationary states were obtained by solving the pseudo-stationary system of equations representing the state, costate and design equations. The main advantages of this method were that it requires no additional globalisation techniques, that it blends nicely with a previously existing pseudo-timestepping method for the states only and that a

preconditioner can be used for convergence acceleration which stems from the reduced SQP methods.

The adjoint-based optimisation has performed in multi-point cases as well. The author in [154] describes a method for drag optimisation of airfoils in three different operating conditions. Also Reuther along with Jameson et al [155] used the adjoint method to perform constrained multi-point optimisation. He developed a shape optimisation method for aerodynamic design problems involving complex aircraft configurations and multiple design points that are subject to geometric constraints. Jameson worked on multi-point design also with Leoviriyakit [156] and developed a method for wing section and planform optimisation at a specific flight condition, which searches for a single wing shape that performs well over a range of flight conditions. A new cost function was defined as the weighted sum of cost functions from a range of important flight conditions. Results showed improvement in performance measures such as drag divergence Mach number and the lift-to-drag ratio over a range of Mach numbers is significantly greater. The applied mathematical fundamentals of numerical multi-objective constraint optimisation were briefly outlined in a lecture [157] given by Wild and Brezillon last year. A description of the design targets and constraints for high-lift wings was given, followed by a detailed analysis of the properties of the flow calculation for the use within optimisation and the suitability of optimisation algorithms for this type of design problem.

Nemec and Aftosmis [158] used a discrete adjoint method for embedded-boundary Cartesian meshes treated as a rigid structure. The formulation was based on the linearisation of a simple geometric constructor, which decoupled the computation of shape sensitivities of the surface triangulation from the cut-cell sensitivities. As a result, the method is well suited to CAD-based optimisation using parametric solid models. Detailed verification studies of gradient accuracy were presented for several two- and three-dimensional shape optimisation problems.

Giannakoglou et al [159] developed a hybrid inverse method for airfoil design. The method couples genetic algorithms (GAs) with an adjoint gradient-based optimisation method in a parallel environment. The GA-based optimisation method was used as the means to drive the solution in the area of the optimum while the gradient-based method

refines the solution in a few iterations. The parallelisation of the GA and flow code significantly reduces the computation cost.

An adjoint algorithm for efficiently incorporating the effects of mesh sensitivities in a computational design framework was introduced in [160]. The method eliminated the need for explicit linearisations of the mesh movement scheme with respect to the geometric parameterization variables, an expense that has hindered large-scale design optimisation for practical applications. The effects of the mesh sensitivities can be accounted for through the solution of an adjoint problem equivalent in cost to a single mesh movement computation, followed by an explicit matrix–vector product whose cost scales with the number of design variables and the resolution of the parameterized surface grid. The methodology augmented the use of adjoint solely for the flow field leading to dramatic computational savings. The accuracy of the implementation was established through several sample design optimisations.

In hydrodynamics, based on Jameson’s [6] work, Cowles and Martinelli [161] described a control-theory (continuous) approach to sensitivity analysis for incompressible, turbulent viscous flows and applied this approach to match target pressure distributions (inverse design) on finite span wings and sails. Tahara et al [162] used a Navier-Stokes code, with finite-difference gradients, for CFD-based design of the bow bulb on a surface combatant. The objective of that study was to minimise the downstream vorticity in the vicinity of the bulbous bow where no free surface effects were considered. Dreyer and Martinelli [163] utilise a continuous-adjoint approach for target pressure matching of propulsor configurations using the pseudo-compressible Euler equations in a rotating frame of reference.

Finally, an overview of the design process is provided by Jameson and Vassberg [9], covering some of the typical decisions that a design team addresses within a multi-disciplinary environment. A review on how the aerodynamic design process has been advanced, first with the improving capabilities of traditional computational fluid dynamics analyses, and then with aerodynamic optimisations based on these increasingly accurate methods, is presented. The computational cost of the adjoint method is also compared with that of other approaches in shape optimisation.



### 2.3.4 Other applications of the adjoint method

The adjoint method has been used in many cases to solve engineering problems that are not related to shape optimisation. In this subsection some of these applications are summarised.

During his research Giles broached the importance of the adjoint solution in analysing the numerical error in integral functionals, such as lift and drag, which are often the quantities of most interest in engineering computations. Along with Suli et al he wrote a report on the error analysis for finite element approximations of the incompressible Navier-Stokes equations [164] in which they established the importance of the superconvergence property. They defined this property as the convergence in which the order of accuracy of the functional increases twice as quickly as the order of the polynomial finite element function space. They derived estimates for the error in a variational approximation of the lift and drag coefficients of a body immersed into a viscous flow governed by the Navier-Stokes equations. The variational approximation was based on computing a certain weighted average of a finite element approximation to the solution of the Navier-Stokes equations. Their main result was an a posteriori estimate that puts a bound on the error in the lift and drag coefficients in terms of the local mesh size, a local residual quantity and a local weight describing the local stability properties of an associated linear dual problem. The weight was approximated by solving the dual problem numerically. The error bound becomes thus computable and can be used for quantitative error estimation; they applied it to design an adaptive finite element algorithm specifically for the approximation of the lift and drag coefficients.

In 1998, they followed this with a paper in which they developed a closed form solution for the adjoint quasi-1-D Euler equations, with and without shocks, and showed excellent agreement with numerical computations [118]. In that paper they also showed how the error in the 'lift' (the integral of pressure along the quasi-1-D duct) could be improved by correcting the leading order effects of the truncation error in the finite volume method; the same idea has been outlined in a short paper for a conference honouring Earl Murman [165]. That paper outlined how the solution of an appropriate dual problem can be used to estimate the residual error of a nonlinear functional in CFD computations. The error estimates can be used either to obtain better approximations to the functional itself, or to drive grid adaptation with the aim of achieving the most accurate answer possible for a given level of computational effort. The finite volume analysis

showed that on unstructured grids discrete conservation is crucial in gaining one order of accuracy relative to the order of the local truncation error. However, the analysis outlined made the assumption that the gradient of the dual solution is bounded. This may not be true for the Euler equations along the stagnation streamline and so additional analysis may be required. The a priori finite element error analysis reveals an interesting superconvergence property showing that the order of accuracy of the approximate linear functional is twice that of the solution itself. The lack of a similar result for the finite volume analysis indicated a significant advantage for finite element methods, but the advantage only appeared when using methods which have better than second order accuracy.

The realisation that the error correction was much more easily constructed by interpolating the numerical finite volume solution to obtain an approximate analytic solution, and then using this to evaluate the residual error and thereby form an inner product with an approximate adjoint solution to correct the computed value for the functional has been the central idea behind a sequence of papers [166, 167] in which they developed the analytic theory and demonstrated super convergent correct functionals for a variety of problems, starting with the 1-D and 2-D Poisson equations and progressing to the quasi-1-D Euler equations. In each case they used the adjoint correction to achieve fourth order accuracy for functionals which would otherwise have been only second order accurate. This is achieved through error analysis that uses an adjoint PDE to relate the local errors in approximating the flow solution to the corresponding global errors in the functional of interest. Numerical evaluation of the local residual error together with an approximate solution to the adjoint equations may thus be combined to produce a correction for the computed functional value that yields the desired improvement in accuracy.

The numerical results demonstrated the effectiveness of the technique applied to a second order finite volume approximation of the quasi 1-D Euler equations. When the flow is smooth, the error in the integrated pressure is fourth order, when there is a shock, it is third order. The theory is equally applicable to the Euler and Navier-Stokes equations in multiple dimensions. Three important issues were addressed so that similar results can be obtained for airfoil and aircraft super convergent lift estimates through adjoint error analysis. The first issue is the treatment of curved surfaces to achieve fourth order accuracy for corrected functional such as lift and drag, it is likely that smooth curved sur-

faces will need to be approximated in a way which ensures continuity in the surface normal as opposed to the use of simple linear or bi-linear faces. The second issue is the resolution of singularities, the adjoint flow solution in two dimensional airfoil applications has an inverse square root singularity along the incoming stagnation streamline and this will need to be well resolved. The final issue concerns unstructured grid calculations which are needed for complex applications. They depicted the need for the approximate solution to be sufficiently smooth that the error is of the same order as the error of the solution itself. To achieve this on unstructured grids where the solution error has significant high frequency content may require the use of multi-dimensional smoothed cubic splines.

In 2001 along with Muller [168] they extended the method for mesh refinement based on adjoint sensors. The advantages of the proposed method was better grid convergence for an integral functional than sensors without adjoint weighting or smoothing. Another application of the adjoint equations in aerodynamics can be found in Joslin et al [169]. They used optimal control and the adjoint method in order to develop a methodology for transition delay.

Very recently Giles [170] used the adjoint method for an application in finance. This application is entirely different from the ones described in the thesis but it is included here to show the generality in the use of the adjoint method. The adjoint method was used to accelerate the calculation of Greeks by Monte Carlo simulation. The method calculates price sensitivities along each path; but in contrast to a forward path wise calculation, it works backward recursively using adjoint variables. Along each path, the forward and adjoint implementations produce the same values, but the adjoint method rearranges the calculations to generate potential computational savings. The adjoint method outperforms a forward implementation in calculating the sensitivities of a small number of outputs to a large number of inputs.

## **2.4 CAVITATION AND OTHER RELATED STUDIES IN ENGINE DESIGN**

In this section research in cavitation and optimisation methods for cavitation reduction and delay which will be used as a connection between the previous sections and the work reported in this thesis are discussed.

The flow through diesel fuel injector nozzles is important because of the effects on the spray and the atomisation process. From the experimental point of view, visualisation of the flow in realistic multi-hole Diesel nozzles has only recently been possible. Arcoumanis et al [171] succeeded in visualising the flow in one of the holes of a modified production conical sac nozzle; the necessary modification involved the removal of some material and the fitting of a quartz window replacing the injection hole. The whole process involved very accurate machining in order to make sure that the hole characteristics remained the same. The rest of the closed-loop flow test-rig comprised a CCD camera with high magnification lenses and a common rail system. Due to the fact that the experiments were steady-state with a constant needle lift only moderate injection pressures could be attained. Two needle lifts and various back pressures were tested; the nozzle was submerged, in order to make the obtained results directly comparable with previously obtained ones with the enlarged model. The comparison of the measured discharge coefficients showed that for high needle lift there is better agreement between the large-scale and the real-size nozzle. It confirmed also for the real-size nozzle that after the onset of cavitation, the discharge coefficient drops with increasing cavitation number, reaching asymptotically a minimum value. Reynolds number was not found to play a role under cavitating conditions. Another important finding was that string cavitation could be seen in the real-size nozzle too, but only in the high needle lifts case. Visual comparison of the various flow regimes between the two nozzles revealed similarities but also distinct differences. In the real-size nozzle and for low Reynolds and cavitation numbers the observed structures collapsed inside the hole, which is in contrast to the large-scale nozzle's case, by which cavitation structures would always exit the hole. Nevertheless, the effect of increasing cavitation number was dramatic in both cases, with cavitation in the real-size exiting the nozzle hole in misty form. Finally, another important finding was that the observed structure sizes did not scale with hole size; in the real-size they occupied relatively larger volume fraction, a point supporting the claim that cavitation as a phenomenon cannot be scaled.

Complementary to the above investigation, Afzal et al [172] performed similar experiments with the same test-rig, but this time they tested and visualised the flow not only in a mini-sac but also in a similar enlarged transparent VCO nozzle. The comparison was of particular interest, as in VCO nozzles higher turbulence levels are realised. Moreover, single-phase CFD simulations of the flow inside the two nozzles were performed,

in order to gain insight into the effects of needle lift and eccentricity on turbulence and other flow field quantities. The comparison between the mini-sac and VCO flow fields revealed interesting differences regarding cavitation characteristics. For low needle lifts in the VCO and with increasing cavitation number no stable film would form at the upper part of the holes. Moreover, for high lift cases, a film could indeed appear but it was much more unstable than the corresponding one in the mini-sac's case. Overall, it is fair to say that the cavitating flow inside the VCO nozzle was found to be much more unstable, and there were two possible reasons given for this. The first one was increased turbulence and the second one was different string cavitation characteristics. In the VCO nozzle for low needle lift string cavitation did not occur in the very restricted sac area but between the needle and the hole entrance, with an axis almost parallel to that of the hole. Adding to that, it was much more unstable in nature for both lifts, which through its interaction with hole cavitation caused complete or partial destabilisation of any formed cavitation film, depending on whether it was low or high needle lift case.

Concluding on enlarged model investigations, the study of Roth et al [173] will be discussed, which was focused not only on developed cavitation but also on its transient evolution. In this study the above described test-rig of Arcoumanis et al was used in order to obtain quantitative information about the flow field inside the nozzle. The refractive index matching technique together with an LDV system were utilised for taking detailed velocity measurements inside the enlarged mini-sac nozzle under single-phase and cavitating conditions. Additional to these measurements, still imaging and high-speed video were used to visualize the cavitating flow in both the mini-sac and the VCO nozzles. Quite interesting findings were made in this study; the importance of string cavitation became even more evident, due to the fact that cavitation strings were found to induce hole cavitation, as they were seen to transport bubble nuclei from one hole, where cavitation would have already started, to an adjacent one which wouldn't be cavitating up to that time instance. Hole-connecting strings could be seen for both high and low needle lifts in the mini-sac's case, whereas for the VCO they could be only seen for high lifts. Cavitation was found to initiate also at the side corners of the hole inlet. From the LDV measurements it was concluded that turbulence levels increased in the lower part of the injection hole as the cavitation number was increased. Nevertheless, towards the hole exit this increase of turbulence was found to die-out and almost level with the lower cavitation number cases.

From the computational point of view, modelling the nozzle flow is complicated by the presence of cavitation inside the nozzles. Chen [174] developed a few different models of cavitating flow. In one case he closed the hydrodynamic equations by assuming that the growth phase of the bubbles occurred at the vapour pressure and that the substantial derivative of density could be given by an empirical relation during the collapse phase. The pure liquid phase was treated as incompressible. This model was applied to a sharp nozzle and appeared to give reasonable results but was not fully explored. Marcer and LeGouez [175] developed a multiphase Navier-Stokes model using an improved VOF type interface tracking method able to calculate the liquid-vapour interface evolution, taking into account the mass transfer process at the interface cavity. The model allows numerical description of the onset and development of cavitation within Diesel injectors. The results obtained confirm the findings of other investigations in that the nozzle discharge coefficient depends mainly on the cavitation due to the volume occupied by the vapour structures which tends to reduce the flow rate. The discharge coefficient gradually drops with increasing cavitation number reaching asymptotically a minimum value. Some examples of calculations performed with the multiphase code EOLE based on an improved VOF method taking into account mass transfer process, evidence the great capacities of the model to simulate cavitation features in Diesel injectors. Numerous validations on measurements have been already carried out on parameters such as:

1. discharge coefficient,
2. injection velocity,
3. visualisations of the cavitation shape,
4. influence of the inlet shape of the channel (straight entrance, rounded entrance).

The paper of Yuan et al [176] deals with the numerical simulation of cavitation phenomena inside injector nozzles. The numerical approach combines the Volume-of-Fluid technique (VOF) with a model predicting the growth and collapse of bubbles. The effect of the liquid quality in the cavitation inception is extensively studied. The numerical study of a symmetric injection nozzle has identified several complex phenomena which certainly occur inside fuel injector nozzles. Numerical results indicate that the overall extension of the cavitation region depends on the liquid quality and the nozzle pressure difference, i.e. the higher nuclei concentration and the bigger nuclei radii as well as the lower pressure at the nozzle exit cause larger overall extension of the cavitation. The numerical study also demonstrates that the rounding of the nozzle inlet lip tends to inhibit the overall extension of the cavitated region. This was used as an indication in the

present study that led to the parameterisation of the radius of curvature. At the same time, the numerical simulations point out, that the flow separation in injector nozzles may be caused due to the sharpness of the nozzle inlet lip as well as by the cavitation process. The separation formed by the primary single flow influences the extension of the two-phase region. To simulate the interaction between the flow separation and the flow cavitation more accurately, the turbulence model needs to be improved.

Another paper that gave hints for the nozzle parameterisation developed in this study was the one of Schmidt et al [177]. He uses a two dimensional, two-phase, transient model of cavitating nozzle flow to observe the individual effects of several nozzle parameters. The injection pressure is varied, as well as several geometric parameters. Results are presented for a range of rounded inlets, from  $r/D$  of  $1/40$  to  $1/4$ . Similarly, results for a range of  $L/D$  from 2 to 8 are presented. Finally, the angle of the corner is varied from  $50^\circ$  to  $150^\circ$ . An axisymmetric injector tip is also simulated in order to observe the effects of upstream geometry on the nozzle flow. The injector tip calculations show that the upstream geometry has a small influence on the nozzle flow. Results show that the cavitation inception is reduced in the rounded geometries. Schmidt et al [178] photographs revealed that the liquid-vapour interface is smooth near the inlet corner and becomes rough and convoluted near the end of the cavitation region. The numerical predictions of density agreed with the experimental photographs. The numerical results also showed the pressure field, streamlines, and exit momentum. Experimentally, the occurrence of cavitation at the inlet corner was found to be similar to that of axisymmetric nozzles. The occurrence of separation at the nozzle exit was found to be a function of Reynolds number for low cavitation numbers and a function of cavitation number for high Reynolds number. This is a logical consequence of the fact that cavitation number is the dominant parameter for cavitating flows and Reynolds number is the dominant parameter for non-cavitating nozzles.

From the work of Arcoumanis and Gavaises [179] it was concluded, after discussing a large amount of experimental and modelling investigations, that although numerous theoretical attempts have been made to simulate the flow inside the nozzle holes of Diesel injectors, there is still a need to develop a method which can predict most of the characteristics of cavitation more accurately. This led to the work of Giannadakis et al [180]. They presented a recently developed cavitation model for automotive fuel injection systems which is based on the Eulerian - Lagrangian approach. The model, which is

presented in more detail in [18], incorporates many of the fundamental physical processes assumed to take place in cavitating flows such as bubble formation through nucleation, momentum exchange between the bubbly and the carrier liquid phases, bubble growth and collapse due to non-linear dynamics, bubble turbulent dispersion and both bubble turbulent/hydrodynamic break-up. The effect of bubble-to-bubble interaction on momentum exchange and during bubble growth/collapse is also considered. In addition, the model incorporates an approach accounting for bubble motion in Eulerian grids with cell size comparable to that of the bubbles. This effect is accounted for through an improved conservative method that calculates the volume fraction of bubbles which extend to more than one cells, and an improved impingement methodology that considers the actual point of contact of the bubble surface with the solid boundary. By using the model in a solved single-phase flow field under tension, it has been possible to predict significant reduction of the volume of the liquid under tension and in its flow rate compared to the single-phase simulation, which is consistent with the reduction of the nozzle discharge coefficient observed under cavitating conditions. An increase of the average velocity of the liquid emerging from the nozzle is also calculated, in agreement with available experimental data. This model is the one used for the presented research. An even more detailed approach would be to solve a separate conservation equation for the bubble number density. This approach was employed in the cavitation model of Matsu-moto et al [181].

There have been also many theoretical studies on vortex cavitation inception; Hsiao et al [182] developed a model to predict cavitation inception which was based on improved bubble dynamics and the interaction of bubble trajectories with the flow field induced by an ideal Rankine vortex. With their model they were able to predict cavitation event rates based on both an “optical inception criterion” and an “acoustical inception criterion”. They found scaling effects to be important due to the different behaviour in terms of both growth and acoustic response of various nuclei sizes.

#### **2.4.1 Work in internal flows and engine components optimisation**

As it was mentioned in the previous section it's only been recently that engineers managed to model the flow inside nozzles. Moreover, performing computational optimisation in these geometries is very limited. So far, most efforts on the adjoint-variable method in CFD shape optimisation have focused on the aerodynamics of airfoils, which



involve very simple boundary conditions, especially the ones in the far field. With simpler boundary conditions for the flow equations, boundary conditions for the adjoint equations are easier to formulate. There is some work in optimisation where the cases are similar to the ones of this thesis but in very simple geometries. In this section all the relevant research done in the area of the thesis is stated.

Soto and Löhner [146] proposed a methodology to solve design problems using an incomplete-gradient adjoint formulation based on the continuous approach to sensitivity analysis where only the adjoint on the boundary of the domain is computed. The continuous adjoint formulation for incompressible Euler design problems, and a scheme to compute the sensitivities which does not depend on the CAD representation, were derived. To do this, an innovative pseudo-shell surface parameterisation scheme for the three-dimensional problems was introduced. The procedure is not only very cheap from the computational point of view, but also produces smooth, singularity-free surfaces, a highly desirable characteristic in any optimisation methodology. The scheme has also the important advantage that the perturbation of a single point produces a smooth perturbation on the entire surface, allowing the finite difference gradients to be approximated in an accurate manner. Several examples indicate the present scheme yields proper results without having to incur the cost of a complete adjoint solution. Two of the cases examined in that paper were relevant to the present studies. The objective of these examples was to maximise the minimum pressure over a hydrofoil at a fixed lift or over the forward part of a 2-D hull, in order to avoid possible cavitation. This type of optimisation objective is often encountered in hydrodynamics where cavitation is always a concern. This work was extended for the complete adjoint equations in incompressible flows [147] and in the RANS equations [149] where again they present some cavitation reduction cases. In [147] also an innovative and very fast volume mesh movement algorithm was developed, which allows including the interior point contributions to the gradient.

An adjoint-variable method was developed to perform gradient-based shape optimisation of internal flow passages in which the flow is governed by the steady-state, three-dimensional, compressible Navier-Stokes equations Choi et al [183]. He dealt with internal flows optimisation, the boundary conditions arising in these cases and suitable cost functions for engines like reduction of pressure drop etc. The geometry optimised was a simple generic internal flow passage. Also Lim et al [184] used the adjoint

method to optimise this time a two-dimensional diffuser using a simple linear parameterisation. He also examined the effect of turbulence in the overall optimisation process. Recently, Kaminski et al [185] used an automatically generated discrete adjoint Navier-Stokes code in the automotive design process. The design application consisted in the optimisation of the topology of a duct for the cabin ventilation. The design approach was based on a descent algorithm, which relied on the gradient of dissipated energy with respect to the duct topology. This gradient was provided by the adjoint of the solver, which is generated from the solver's source code by the automatic differentiation tool TAF. All the grid points were used as parameters for the optimisation problem. Results showed that the usage of TAF allows automating the maintenance of the derivative code and, hence, the update process of the entire system to new releases of the nominal solver. This in turn reduces the delay between model development and design applications, i.e. the overall efficiency of the design cycle can be increased.

Other methods apart from the adjoint method have been used for shape optimisation in engine components. For example Trigui et al [186] investigates the problem of IC Engine shape optimisation using the finite difference method where the discharge coefficient is used as the design objective function, while the total angular momentum flux defines the constraint function. Madsen [187] developed a method for optimisation of internal flow devices based on the parameterisation software ODESSY. Again, in this case simple diffuser geometries were examined with objectives in the interests of automotive engineers. Brewer et al [188] dealt with the problem of cavitation delay in various geometries like open water propeller and rotary blood pump. Again the problem of cavitation is dealt in the single face through the increase of the pressure where negative values appear. For the gradient calculation automatic differentiation was used. Optimisation for a medium speed direct injection diesel engine was conducted by Bergin et al [189]. The angular alignment, nozzle hole size, and injection pressure of each set of nozzle holes were optimised using a micro genetic algorithm. The design fitness criteria were based on a multi-variable merit function with inputs of emissions of soot, NO<sub>x</sub>, unburned hydrocarbons, and fuel consumption targets. Penalties to the merit function value were used to limit the maximum in-cylinder pressure and the burned gas temperature at exhaust valve opening. Results showed significant decrease in NO<sub>x</sub> and soot from the baseline case and improvement in fuel economy. An illustration of a modified prioritisation of the emission targets in optimisation studies, which may be crucial to attaining aggressive emission standards, was presented in [190]. A direction for heavy-

duty diesel engines to meet future stringent emission mandates is also presented in the same paper.

## 2.5 BACKGROUND

The work reported in this thesis is a continuation of the research work of the author in the area of aerodynamic optimisation. The scope of the optimisation in that case, was again as in the case of the thesis, shape modification for performance improvement. The basic steps of the optimisation procedure overall are the choice of the most suitable shape parameterisation and the determination of the design variables on the multidimensional design space. It is also the bounding limitations of the design variables when geometrical constraints are considered as well as the formulation of the most proper objective functional for each case examined and finally the mathematical assumptions that have been adopted during the adjoint problem derivation procedure (i.e. costate variables, etc).

In the aforementioned research work a continuous approach of the adjoint technique has been developed and applied to a finite volume CFD solver for compressible flow using high resolution upwind schemes [191]. The code could handle both structured and unstructured meshes and was based on a time-marching, primitive variable, finite-volume method for the solution of the Euler or Navier-Stokes equations. The Roe flux-difference splitting was used for the calculation of the inviscid fluxes. Second order accuracy was achieved through the implementation of the MUSCL extrapolation scheme. Viscous fluxes are discretised using a central scheme for the structured cases and methodologies from finite elements for the unstructured. The Barth-Jespersen limiter was used to limit the required flow variable gradients. The Spalart-Almaras one equation model, which also contains a laminar to turbulent transition mechanism, provided the location of the transition point is used for turbulence modelling. In the solution of the adjoint system of equations the turbulent viscosity was considered constant. This is a realistic assumption if slight changes of the geometry are considered during the iterations. The solution in each time-step was updated by means of a pointwise-implicit Gauss-Seidel iterative scheme. The CFD code had the ability of capturing shock waves, separation, vortices etc. In Figure 2.1 the Mach distribution of an SCT wing in zero angle of attack and  $M_\infty = 2$  is represented. The same techniques that are used for the

direct solver are also employed for the numerical solution of the adjoint system of equations (on the same grid at each optimisation cycle).

For the parameterisation of the geometries Bezier polynomials were used for the 2-D cases (airfoils, cascades etc.) while for the 3-D cases NURBS surfaces were applied. The Bezier Polynomials were formulated with clipping of the control points at leading and trailing edge for the imposition of the corresponding aerodynamical constraints. More detail about these parameterisation methods exists in section 3.5. Figure 2.2 shows the parameterisation of a SCT wing using NURBS.

The implementation of an efficient and very fast algebraic mesh generator for the required structured C-type stretched grids or advancing front for the unstructured ones, into the optimisation code eliminated the need for grid modification at least in the 2-D case. In the 3-D case the grid modification was realised by a spring method based on interpolations between the previous and current position of the boundary under modification. The same method is used for the present study and is explained in section 3.6.

At first the method was applied in inviscid flow problems and the continuous adjoint equations were calculated having as constraints the Euler equations. Two different problems, having two different cost functions, have been solved using this approach:

1. The problem of inverse design of an airfoil knowing the pressure distribution along its boundary.
2. The problem of the drag coefficient minimisation of an airfoil or a wing lift involving lift and pitching moment coefficients, appropriately weighted for the optimisation.

In [154] the problem of airfoil design for multi-point performance optimisation is addressed. In particular, drag minimisation is sought for the RAE2822 airfoil in three different operating points, while a prescribed lift coefficient at each point has to be maintained. The optimisation method was also extended for the design of aerodynamical shapes taking into account viscous effects and turbulence modelling. In this case the Navier-Stokes equations are treated as constraints for the derivation of the adjoint equations. Both cases were also considered for viscous flows some representative results are shown in Figure 2.3 and Figure 2.4. However, the known pressure distribution de-

sign is not straight forward in this case (see section 3.8 for details). Finally in Figure 2.5 the convergence history of both cases is presented. More details also for the above approaches can be found in [192].

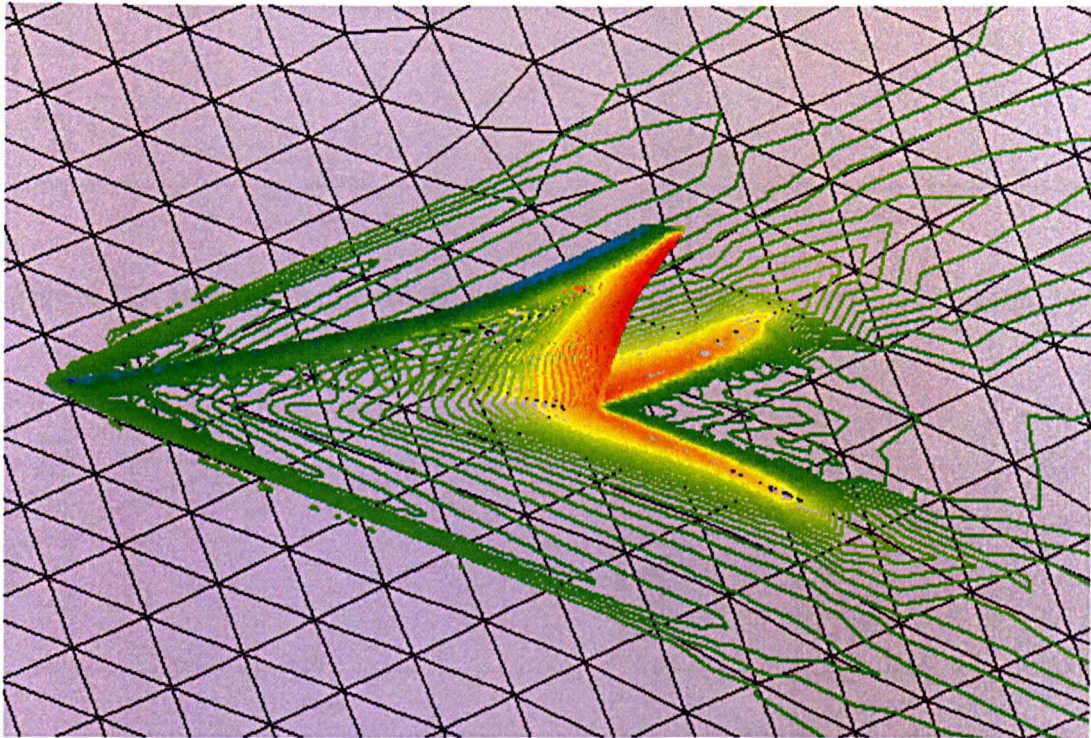


Figure 2.1: Mach distribution of EUROSUP in zero angle of attack and  $M_\infty = 2$ .

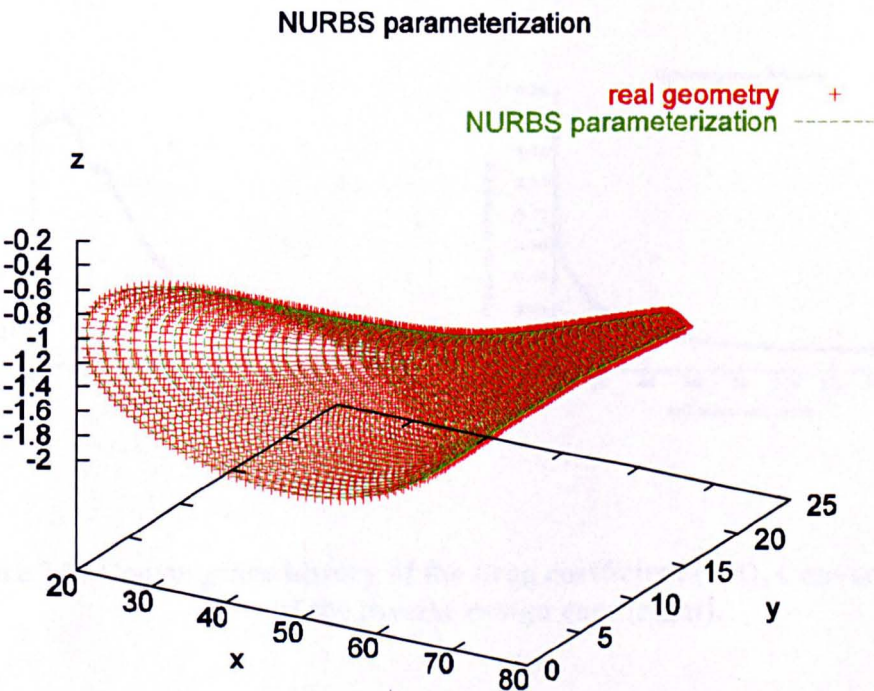




Figure 2.2: The initial NURBS parameterisation of EUROSUP

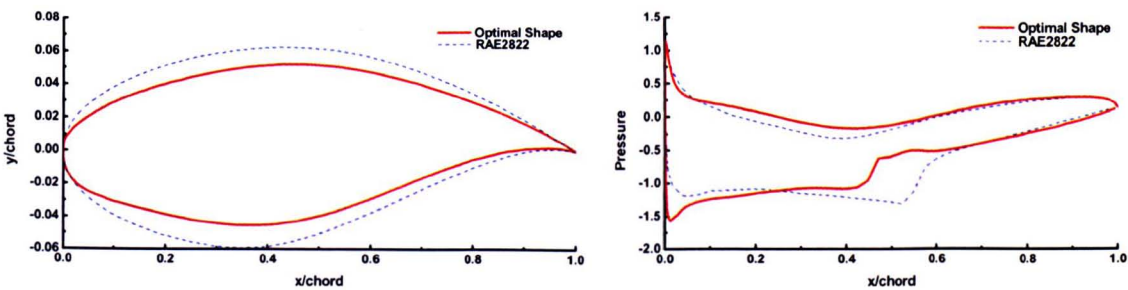


Figure 2.3: Derived profile shape vs. the original (left) pressure distribution vs. the experimental one (right) of the *RAE2822* airfoil

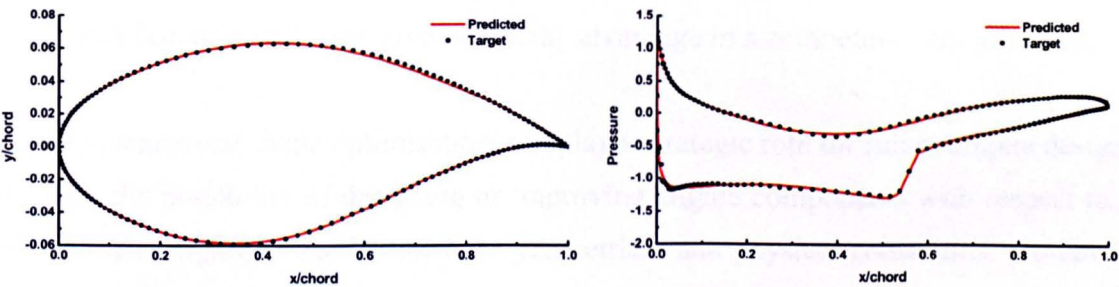


Figure 2.4: Derived profile shape vs. the original (left) pressure distribution vs. the target one (right) of the *RAE2822* airfoil

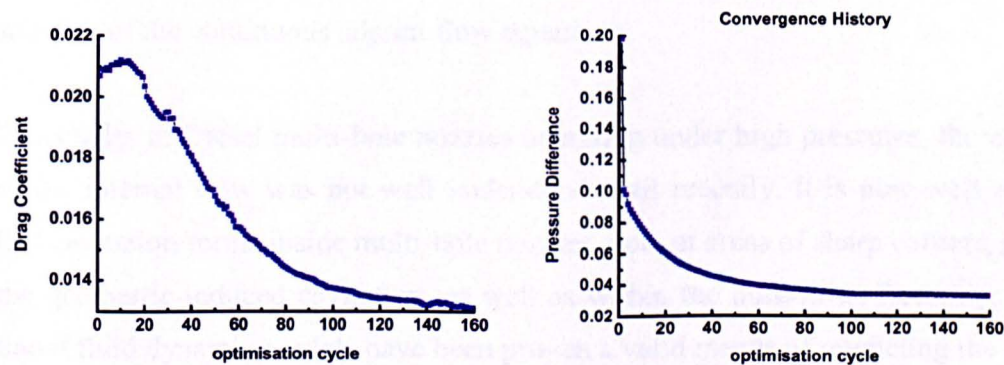


Figure 2.5: Convergence history of the drag coefficient (left), Convergence history of the inverse design case (right).

## 2.6 CONCLUSIONS

Numerous investigations about shape optimisation methods and their effect on cavitation inception were discussed in this chapter. There are several conclusions that can be drawn out of the work presented here and how these affected the choices made during the present research. It is obvious from the literature that automated design has elevated the aeronautic industry. On the other hand, in the automotive industry, there is the need for more efficient method for engine component design. In the automotive industry experiment is still the main designing tool. Nevertheless, it's not at all likely that repeated trial in an interactive design and analysis procedure can lead to a truly optimum design. Using an automated optimisation method, not only can designs be rapidly evaluated but directions of improvement can be identified as well. Possession of techniques which result in a faster design cycle gives a crucial advantage in a competitive environment.

Detailed numerical shape optimisation will play a strategic role for future engine design. It offers the possibility of designing or improving engine components with respect to a pre-specified figure of merit subject to geometrical and physical constraints. However, the extremely high computational expense of straightforward methodologies currently in use prohibits the application of numerical optimisation for industrially relevant problems. Optimisation methods based on the calculation of the derivatives of the cost function with respect to the design variables suffer from the high computational costs if many design variables are used. However, these gradients can be efficiently obtained by solution of the continuous adjoint flow equations.

Especially, in Diesel multi-hole nozzles operating under high pressures, the complexity of the internal flow was not well understood until recently. It is now well recognised that cavitation forms inside multi-hole nozzles, both at areas of sharp corners, producing the geometric-induced cavitation, as well as within the bulk-flow. Recently, computational fluid dynamic models have been proven a valid means of predicting the formation of cavitation and its further development. Controlling cavitation greatly depends on the nozzle geometric characteristics and operating conditions. In most diesel injectors operating at high pressure exceeding 1500bar, cavitation initiates at the entrance of the nozzle hole. In most cases, the cavitating structures finally reach the nozzle hole exit. Although the effect of cavitation on the spray formation still remains largely unknown, it is well recognised that uncontrolled cavitation induces spray-to-spray and cycle-to-cycle

variations; that, in turn, increases engine exhaust pollutant emissions. By changing the shape of the hole inlet (curvature, conicity, needle seat angle) cavitation formation in the hole can be affected. The discharge coefficient is a valid mean to measure the cavitation inception. These observations helped in the choices made for the development of the optimisation method used in this study. The developed method consist of designing nozzle hole shapes that reduce or even eliminate the local minimum/negative pressure region formed at the entry of hole-type nozzles. In the subsequent chapters the development and application of such an optimisation method will be presented.



## *Chapter 3*

# *FORMULATION OF THE OPTIMISATION PROBLEM*

### **3.1 INTRODUCTION**

In this chapter the basic idea of shape optimisation is presented in detail. The mathematical definition of an optimisation problem is given and the elements of this problem are specified; these include parameters, cost function and constraints. The solution of the optimisation problem is obtained through a gradient based method. This means that gradient information of the cost function is required to determine the search direction towards the optimum. For the calculation of the required gradients the continuous adjoint method has been used. The solution procedure is described thoroughly in this chapter. The features of the problem are described together with the parameterisation methods used and developed for the needs of this study and the automatic grid modification method along with the definition of the proper cost function of the problem. Also the mathematical model for the fluid flow is presented. The Navier-Stokes equations have been used as constraints for the formulation of the optimisation problem. The continuous adjoint method has been used as the means to calculate these gradients with respect to the design parameters. Extended analysis of the use of the method exists in this chapter. The adjoint system of equations is formulated along with the boundary conditions. The difficulties of the implementation for different cost functions are discussed along with the way to overcome them. Finally the gradients are formulated for the different cost functions used in this study.

## 3.2 MATHEMATICAL FORMULATION OF GENERAL OPTIMISATION

### PROBLEM

Every parametric optimisation problem which involves quality criteria can be represented by an equivalent mathematical minimisation or maximisation problem of a real function involving one or more variables. For the mathematical representation of the optimisation problem a vector of real parameters  $\vec{D} = (X_1, \dots, X_n) \in \mathbb{R}^n$  is introduced as well as a real function or functional  $I(\vec{D}): A \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$  (cost or objective function). We are trying to minimise (maximise) this function; meaning finding the point of its domain that corresponds to the functions minimum (maximum) value. The following formula proves the equivalence of minimisation and maximisation problems:

$$\arg \max_{\vec{D} \in A} I(\vec{D}) = \arg \min_{\vec{D} \in A} (-I(\vec{D})) \quad (3.1)$$

So for the remaining of the thesis all problems will be referred to as minimisation problems. As it will be discussed in the following sections, even when we seek to modify the geometry in order to achieve the maximum discharge coefficient  $C_d$ , the case is handled as a minimisation case through the proper definition of the corresponding cost function.

Another element which may arise in an optimisation problem is some constraints imposed by either the physics of the problem or the geometry of the shape, say:

$$J(\vec{D}) \begin{cases} \geq \\ = \\ \leq \end{cases} 0 \quad (3.2)$$

Every equality constraint introduced in the optimisation problem reduces the number of optimisation parameters by one, if it is possible to express this parameter as a function of the rest. On the other hand the inequality constraints reduce the magnitude of the search space without changing its dimension. It is quite useful to reduce the inequality constraints in only one type through the formula  $J(\vec{D}) \leq 0 \Leftrightarrow -J(\vec{D}) \geq 0$ .

Since the above elements of the optimisation problem are defined the problem can be expressed as:

Find the point  $\arg\min_{\bar{D} \in A} I(\bar{D})$  given the constraints:  $J_k(\bar{D}) \geq 0, k = 1, \dots, n$   
 $J_l(\bar{D}) = 0, l = 1, \dots, m$

Having defined the optimisation problem we have introduced the target or the solution to the problem without having mentioned the solution procedure or when this solution is possible. These elements are discussed in detail in the next sections. More details concerning numerical optimisation can be found in [35].

### 3.3 OPTIMISATION PROCEDURE

CFD shape optimisation can be defined as a procedure through which the geometry of interest can be modified to fit certain flow characteristics. Introduction of those characteristics into the problem is achieved through the definition of the so-called ‘cost’ function. At the same time, the geometry is modified according to a set of parameters relative to the problem. In deterministic optimisation the cost function is minimised by driving the calculated gradients to zero with the use of a descent method.

The optimisation method developed for the needs of this study is an adjoint-based deterministic method. The proposed inverse design/optimisation method, like any other deterministic method, requires gradient information, i.e. sensitivity derivatives. These are obtained by formulating and solving the adjoint equations. The latter are derived by enforcing the flow equations as constraints to the cost function via the concept of Lagrange multipliers. The continuous approach is adopted in this study so that the resulting formulation, algorithm, and code can be used in conjunction with any CFD code. There two more major advantages of the continuous vs. the discrete approach.

1. The physical significance of adjoint variables and the role of the adjoint boundary conditions are much clearer. Only by constructing the adjoint flow equations one can develop a good understanding about the nature of the adjoint solutions. In aerodynamics where there has been more research on the properties of the adjoint solution the continuous approach helped the understanding of its behaviour. Examples include the continuity at shocks, the logarithmic singularity at a sonic point in quasi-1-D flows but not in 2-D or 3-D (in general) and the inverse square-root singularity along the stagnation streamline upstream of an airfoil in 2-D [119].
2. The adjoint programme is simpler and requires less memory. Because one is free to discretise the adjoint P.D.E. in any consistent way, the adjoint code can be much sim-

pler. However, our experience has been that even when following a continuous approach, it is advantageous to consult the discrete formulation so as to choose an appropriate discretisation for the continuous adjoint equations. It is also generally the case that continuous adjoint solvers require less memory than the fully-discrete codes, but this difference is not substantial if pre-computation and storage of the linearised matrix is avoided when implementing the discrete method.

The problem setup is a difficult task because it can always lead to ill-posed problems. Ta'asan in a VKI lecture [193] demonstrated an analysis which gives a very simple classification of problems based on the asymptotic behaviour of the Hessian using Fourier analysis. It distinguishes between ill-posed and well posed-problems, easy and difficult ones.

The formulation and solution of an optimisation problem using a deterministic method requires some basic initial steps. First an initial geometry is introduced. This geometry is either an arbitrary or a known one which has certain flow characteristics that if modified can conclude to a desired optimised geometry. This geometry has to be given in parametric form. An essential step before the parameterisation of the geometry is the definition of the cost function. The cost function must be defined in such a way that by driving it to zero the optimal geometry, i.e. the one satisfying the required flow characteristics, will be provided. The cost function is the means to quantify the optimum solution so it is really important to define it appropriately for each problem. Further details about the definition of the proper cost function are discussed in the following section of this chapter. At this point some flow constraints can be introduced in the problem. In this study the flow has been considered viscous and assumed incompressible so the Navier-Stokes equations are introduced as equality constraints.

After these steps a parameterisation scheme describing the geometry can be introduced using as parameters the points that control the flow characteristics we want to optimise. For the parameterisation we ought to have in mind certain geometrical constraints that must be enforced to our design and use the knowledge about the behaviour of the flow under certain geometry changes [177]. Furthermore, it is desirable for local geometry regions of importance to be defined by a minimum number of control points so that the search space for the optimum solution is limited. So basically, the set of parameters must include the search space that the design evolves to an optimum solution, while at

the same time constraint it to avoid non physical geometries to be produced. The parameterisation methods developed and used for this study are presented in detail in section 3.5.

After introducing a parameterisation scheme that captures the initial selected geometry and describing the designs of the search space, the initial mesh can be generated. The initial grid is used for the first cycle and then it is modified to provide the grids for the intermediate geometries. The grid in each optimisation cycle is produced via a completely automated re-meshing procedure. Further discussion about the initial grid generation and grid modification during the optimisation process exists in section 3.6.

The completion of the previous steps along with the initialisation of the flow solution leads us to the point where the iterative optimisation procedure can be initiated. The optimisation procedure is constituted by individual cycles for the progressive improvement of the shape as the cost function tends to zero. Each cycle consists of the following steps:

1. Build the geometry using the parameters that control the design.
2. Move the mesh to fit the current geometry.
3. Use the flow solver to calculate the flow field and flow variables and the value of the cost function for the current design.
4. Use the adjoint solver to calculate the adjoint field and the costate variables which are going to be used in the calculation of the cost function's gradient.
5. Calculate the gradient of the cost function with respect to the control parameters/design variables.
6. Use a descent optimisation method to calculate the new parameters by driving the gradients to zero where the optimum is located.

The process is repeated until the gradients and the cost function are driven to zero and the optimum geometry has been produced. All these steps will be discussed in further details in the forthcoming sections. Especially about steps 3 and 4 detailed analysis exists in sections 3.7 and 3.8 respectively. In section 3.7 exists a detailed description of the governing equations used for the calculation of the flow field and therefore for the calculation of the current value of the cost function. In section 3.8 there is a detailed analysis for the extraction of the adjoint system equations along with the boundary condi-

tions. The boundary conditions of the adjoint system of equations strongly depend on the cost function used. The analysis is followed by the formulation of the gradient with respect to the design variables. The extraction of the gradients is described for each cost function used in the present study. The solution procedure of the flow and the adjoint system of equations are described separately in the next chapter. The discretisation and solution of the two systems are cross presented giving details about similarities and differences. The generic transport equation is used as the basis to describe the discretisation and solution process.

The same numerical scheme used for the flow solution is used for the solution of the adjoint system of equations. However, the variation in the turbulence effective viscosity, attributed to the geometry modifications, has been considered negligible. This is a convenient assumption from the numerical point of view and quite realistic for slight changes of the geometry from cycle to cycle.

The numerical solution of the flow and the adjoint equations bear almost exclusively the burden of computational cost. Having in mind that the computational cost for the solution of the adjoint equations is almost equal to the computational time for the direct problem, each design cycle is almost equivalent to two direct solutions. For further reduction of the computational cost, continuation of the flow solution using the solution of the previous optimisation cycle (previous geometry) has been used. This results in significant reduction of the computational time (sometimes even 90%) compared to a flow solution which is initialised for every optimisation cycle.

### **3.4 FORMULATING AN OBJECTIVE FUNCTION**

An objective function accurately quantifies the success of the design modification. To optimise a design it is usually necessary to understand the source of undesired effects. Not until recently, the complexity of the internal flow has become well understood in Diesel multi-hole nozzles operating under high pressures. It is now well recognised that cavitation forms inside multi-hole nozzles producing the geometric-induced cavitation, as well as within the bulk-flow. The latter is a dynamic effect of the vortices formed inside the sac volume and it is still not predictable [194]. Recently, computational fluid dynamic models have been proven a valid means of predicting the formation of cavitation and its further development [180]. In [188] a computational method has been de-

veloped for cavitation delay in hydrofoils. In that case the aim was to delay cavitation inception, and thus, an obvious objective was the increase of the pressure in regions where inception is known to occur. It is also pointed out that it is important to increase also the average pressure over a localised region, so that a continuous solution can be achieved. However, such models have not been applied yet to internal flow designs of interest to the automotive industry, except in very trivial cases [183, 184, 187].

Controlling cavitation greatly depends on the nozzle geometric characteristics and operating conditions. It is common that in diesel injectors operating at high pressure exceeding 1500bar, cavitation initiates at the entrance of the nozzle hole. In most cases, the cavitating structures finally reach the nozzle hole exit. Although the effect of cavitation on the spray formation still remains largely unknown, it is well recognised that uncontrolled cavitation induces spray-to-spray and cycle-to-cycle variations; that, in turn, increases engine exhaust pollutant emissions. The objective of the present study is to develop a method for designing nozzle hole shapes that reduce or even eliminate the local minimum/negative pressure region formed at the entry of hole-type nozzles, and in which geometric cavitation initiates.

The method used for the design is general so the objective can change according to the users needs. However, due to the fact that the optimisation method used in this study is gradient based, the cost function must be continuous and differentiable i.e. a functional. This is not so restrictive in CFD design cases because the flow field is continuous and usually the flow characteristics are defined either in the whole field area or along the boundary of the shape in the case of the drag and lift coefficients in wings or at the flow exit in the case of the discharge coefficient in nozzles. Further discussion about how the choice of cost function influences the adjoint analysis exists in section 3.9 and 3.10 where the boundary conditions are extracted and the gradients are calculated respectively.

In this study two different cost functions have been used, and both result to control of the cavitation inception in the hole inlet, although the approach of each problem setup is different.

In the first approach the problem is being setup as an inverse design problem with a predefined target. The target is a given pressure distribution along the area of the wall of

the nozzle. In particular, it has been recognised that the designer generally has an idea of the kind of pressure distribution that will lead to the desired performance. Thus it is useful to consider the problem of calculating the shape that will lead to a given pressure distribution. Such a shape does not necessarily exist, unless the pressure distribution satisfies certain constraints, and the problem must therefore be very carefully formulated: no shape exists, for example, for which stagnation pressure is attained over the entire surface.

Actually in the case of cavitation control there is no need to predefine the whole wall pressure distribution but the area where the negative pressure appears. So in order to handle the problem of controlling cavitation as an inverse design problem, the cost function formed ensures increase of the pressure in the local areas where negative values appear. A desired target pressure distribution along the boundary of the nozzle is given, and then the following cost function is defined:

$$I_c = \frac{1}{2} \oint_W (p - \bar{p}_t)^2 ds - \oint_W (p - \bar{p}_t) \mu \frac{\partial \mathbf{u}}{\partial \bar{n}} \cdot \bar{\mathbf{n}} ds, \quad (3.3)$$

where  $p$  and  $\bar{p}_t$  represent the current and target pressure respectively,  $\mathbf{u}$  is the velocity vector,  $\bar{\mathbf{n}}$  is the face unit vector along the boundary and  $\mu$  is the effective viscosity.  $W$  stands for wall and is the boundary of the geometry to be modified. The last integral is zero for incompressible flows; nevertheless, it is added to the cost function for the well-posedness of the problem [7]. This complication comes from the method and has to do with the extraction of the boundary conditions that closes the adjoint equation system. Further discussion about this matter follows in section 3.9 where the boundary conditions of the adjoint equations are formulated.

The above approach is really restrictive due to the fact that the target pressure distribution must be known. Usually it can be provided by experimental data from an already known geometry. Although this approach appears to have an evident restriction it is a very good way to validate the optimisation method through the inverse design of a known geometry. This is the reason, as means to validate the developed method, that this case has been included in the present study. The same cost function has been also used in an inverse airfoil design case to prove the generality of the method with respect to different types of geometries/problems.



Clearly, the real power of the method is realised with the development of a suitable cost function for optimised injector design; written as an inverse method, the approach has a more limited practical use. To exploit the power of the method a second approach has been introduced. This approach is based on the fact that cavitation can be controlled through the discharge coefficient ( $C_d$ ) the definition of which reads:

$$C_d = \frac{1}{A_{exit} \cdot \sqrt{\frac{2(p_{in} - p_{out})}{\rho}}} \oint_{exit} \mathbf{u} \cdot \bar{\mathbf{n}} ds, \quad p_{in} \gg p_{out} \quad (3.4)$$

Where  $\rho$  is the density,  $\mathbf{u}$  is the velocity vector,  $\bar{\mathbf{n}}$  is the normal unit vector along the boundary as before  $p_{in}$  and  $p_{out}$  are the pressures at the inlet and outlet respectively and  $A_{exit}$  is the area at the exit. The  $C_d$  takes values from 0 to 1 the last being the less cavitating case [175]. The cost function  $I_c$  introduced for this case takes the form:

$$I_c = \frac{1}{2} (\bar{C}_d - C_d)^2 \quad (3.5)$$

where  $\bar{C}_d$  is the desired value of the  $C_d$  that we want to achieve. The square power in both cost functions ensures the convexity of the cost function and is used for mathematical reasons. As we mentioned in section 3.2, although we seek to maximise the  $C_d$  the problem is set up again as a minimisation problem by subtracting the current  $C_d$  value from the target  $\bar{C}_d$  one.

This is again an inverse design case but now the target geometry is not a known one. The same cost function works for the optimisation case where the user wishes to achieve the maximum  $C_d$  possible so the value of  $\bar{C}_d$  is set equal to 1; this value is not feasible to be achieved. In that case, it is necessary to introduce geometrical constraints through the parameterisation scheme. In this study the goal has always been to obtain feasible  $\bar{C}_d$  values ( $< 1$ ) or to maximise  $\bar{C}_d$  by setting the target value equal to 1.

There are some complications in the application of this design approach that arise from the fact that the cost function is defined on the hole exit cross sectional area whilst the

gradient and the geometry under modification is actually the hole inlet. There is further discussion about these complications and the way to overcome them in section 3.8.

### 3.5 GEOMETRY PARAMETERISATION

An important step in the formulation of any automated design problem is for the designer to explicitly identify an appropriate set of design variables  $D$ . The countless number of different ways to do so makes the decision, of which parameters to pre-assign and which to consider free, far from trivial. The challenge lies in selecting design parameters, which are significant in terms of system performance, yet do not jeopardise the design process by compromising robustness of mesh generation and analysis. The chosen design variables must influence the considered performance measures by means of influencing the CFD solution. This leads to the following categorisation of relevant types of design variables, as originally suggested for structural optimisation by Olhoff and Taylor [195]:

#### Geometrical design variables

1. *Sizing design variables*

Describe cross-sectional geometrical properties, e.g. the width of the duct.

2. *Shape design variables*

Describe shapes of solid surfaces bounding the fluid flow domain, e.g. the shape of an aerodynamic profile of the cross-sectional shape of a duct.

3. *Topological design variables*

Describe topological properties of the flow domain that offer conceptual control of the design domain. These can either be discrete variables, such as a number of guide vanes, or continuous variables, e.g. the control of the distribution of void and solid material in catalytic beds.

#### Loading design variables

Describe position and distribution of inlet (boundary) conditions, e.g. the velocity and angle at which a mixing jet enters the main flow.

#### Material design variables

Describe constitutive properties of the fluid, e.g. the viscosity and heat capacity of bearing lubricants.

### Manufacturing design variables

Describe parameters pertaining to manufacturing processes, which may influence the performance and cost of the construction, e.g. relative surface roughness.

The work presented in this thesis concerns shape design optimisation. For this reason the most interesting class of design variables are the geometrical ones, and in particular shape design variables, whereas topological design variables are not considered.

As we have already mentioned, the pre-described process of representing the actual geometry by a set of control points, called vector of design variables or parameters, is referred to as parameterisation. Through parameterisation the geometry can be modified according to the freedom given to it by the choice of the design parameters. The efficiency of each parameterisation method depends on numerous factors. The geometry under modification, the physics of the problem and the constraints of the manufacturer are some of these factors. A summary of the literature on this area is presented here.

In Jameson's first papers on the use of control theory [6, 196], every surface mesh point was used as a design variable. Using this approach, the complete design space of all airfoil shapes represented by a given number of surface points can be spanned. In theory, this approach would also produce results closer to a true optimum if the optimum shape were to have a high frequency component that could not be captured with other shape functions with less compact support. On the other hand, a problem of this choice is that the smoothness of the shape may not be preserved. This contradicts the assumption of first derivative continuity of the solution in the development of the adjoint formulation. Another way of understanding this problem is that this choice of using each grid point as a design variable admits very high frequencies in the design space. This admittance of high frequencies causes a higher degree of nonlinearity, and, in practice, the higher the degree of nonlinearity the more computationally difficult it is to find an optimum. Jameson developed an implicit gradient smoothing method [6] in order to overcome the difficulty of discontinuity. The gradient smoothing procedure eliminates high frequency components from the design space. Arian and Ta'asan [197] also pre-

sented a simple Fourier analysis which estimates the smoothing of the minimisation process and provides a tool to establish smoothers by preconditioning if needed.

It has already been stated that, the choice of parameters and the parameterisation method are very essential to the overall design and they are really case dependent. The goal is to introduce a set of parameters which include the search space so that the design evolves to an optimum solution and at the same time constraint it to avoid non-physical geometries to be produced.

Though the adjoint method has the advantage of calculating the gradients independently from the number of design parameters, this number is strongly connected to the size of the search space, and thus, retards the convergence of the optimisation process. Furthermore, it is desirable that local geometry regions of importance are defined by a minimum number of control points.

A typical parameterisation scheme, especially in aeronautical design is using Bezier or B-Spline representation in 2-D or a Bezier patch or NURBS (Non-Uniform Rational B-Spline) surface in 3-D [154]. For the nozzle case these schemes though general have not been applied so far, mainly because of their general nature. The shapes produced by such parameterisation cannot be easily constrained to give feasible nozzle shapes that can be actually manufactured. However, for the sake of completeness and generalisation in the present study, the optimisation method has been applied to aerodynamic shapes (airfoils) which are modelled by Bezier-Bernstein curves.

With regards to cavitation control, we are primarily concerned with parameterising only a localised region, such that the overall nozzle design is not changed by changing the flow characteristics of interest. Choi in [183] parameterised a flow passage using a position vector describing the internal wall of the flow passage by a number of circles with given radius, centre and inclination angles. In that case, the passage inlet and outlet diameter and location were constrained and only the surfaces between those two boundaries could change during the optimisation process. Lim in [184] parameterised an asymmetric diffuser using its length, inlet and outlet heights and the opening angle. Trigui et al in [186] used four variables to parametrically represent the intake port surface of an engine. The first two control the shape of the path along which the port sections are swept while the other two control the port cross sections. B-splines are used to

smoothly connect the parameterised parts of the shape. In another approach, presented from Soto and Löhner in [148], the key idea is a scheme in which almost every point on the surface can be chosen as a design parameter. Although the adjoint method is independent from the number of design variables, and the fact that this scheme generates surfaces free of singularities, the search space in this case is extremely large and this would decrease significantly the convergence rate of the optimisation method.

There is no published research so far in optimisation of diesel nozzles. Taking under consideration the above aspects, empirical parameterisation schemes were developed and used to parametrically represent the shape in the cases of the 2-D axisymmetric nozzle and 3-D nozzles. Detail description of these schemes follows in the next sections.

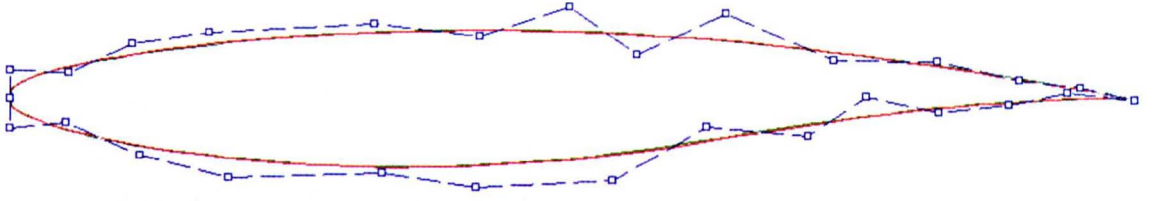
### 3.5.1 Airfoil Parameterisation using Bezier Polynomials

In the present study the aerodynamic shapes are modelled by Bezier-Bernstein polynomial curves of degree ( $n$ ) using ( $n+1$ ) control points. The coordinates along the Bezier curve are written in parametric form as a function of a length parameter ( $t$ ) belonging in the interval. The control points define the ‘convex hull’ polygon for the Bezier curve. Bezier’s control points are the design variables of the optimisation problem. They often subject to specific constrictions-limitations resulting from geometrical constraints just as thickness distribution, common tangents at the leading edge for pressure and suction side, slope and thickness at the trailing edge. For the cases that require a large number of control points (about 50 or more), this parameterisation seems to be very ‘expensive’ for stochastic algorithms or direct gradient-based methods.

The  $x$ - and  $y$ -coordinates of the airfoil boundary are written in parametric form as a function of a length parameter ( $t$ ) belonging in the interval and depending mainly on the control points distribution. Each side of the airfoil is modelled separately by using Bezier polynomials of different degree in general, which for the  $n$ -dimensional vector of the control points  $\vec{D} = (X_i, Y_i), i = 1, \dots, n$ , take the form

$$x(t) = \sum_{i=1}^{n+1} C_i(t) X_i, \quad y(t) = \sum_{i=1}^{n+1} C_i(t) Y_i \quad (3.6)$$

where  $C_i(t)$  are the Bezier coefficients. Figure 3.1 represents a typical airfoil parameterisation.



**Figure 3.1: Parameterisation of initial geometry using Bezier polynomials.**

It is assumed that the  $dt$ -value associated with each boundary node, remains constant regardless of the change in geometry. Consequently, the variation of the  $dx$  and  $dy$ , required in the expressions of the sensitivity derivatives as the variation of the quantity  $\vec{n}ds$  reads:

$$\begin{aligned}\delta(dx) &= \delta\left(\frac{dx}{dt}\right)dt = \sum_{i=1}^{n+1} \dot{C}_i(t)X_i dt, \\ \delta(dy) &= \delta\left(\frac{dy}{dt}\right)dt = \sum_{i=1}^{n+1} \dot{C}_i(t)Y_i dt\end{aligned}\tag{3.7}$$

where the dotted quantities stand for derivatives with respect to  $t$ ,  $\dot{C}_i = dC_i/dt$ .

### 3.5.2 Nozzle parameterisation

As it has already been discussed, one could potentially completely parameterise the nozzle geometry. This would allow for the maximum number of possible designs. Regarding to cavitation control, we are primarily concerned in parameterising only a localised region, such that the nozzle retains certain performance attributes. Cavitation usually occurs in the hole inlet; so practically, we are interested in changing the hole geometry in order to affect the negative pressure distribution appearing in the hole entrance. The parameters affecting the discharge coefficient [177], and as a result the cavitation inception, are:

1. the *radius of curvature*,
2. the *needle seat and relative angles*,
3. the *hole entrance and exit diameters* and
4. the *hole length*.

It is obvious that a general polynomial parameterisation (like Bezier) cannot affect *directly* those parameters and moreover change them without affecting the overall per-

formance of the nozzle. The introduction of a more specified, case dependent, parameterisation scheme is required.

For the needs of this study two different *empirical* parameterisation schemes are developed and used in the case of nozzle design in order to modify the 2-D and 3-D geometries respectively.

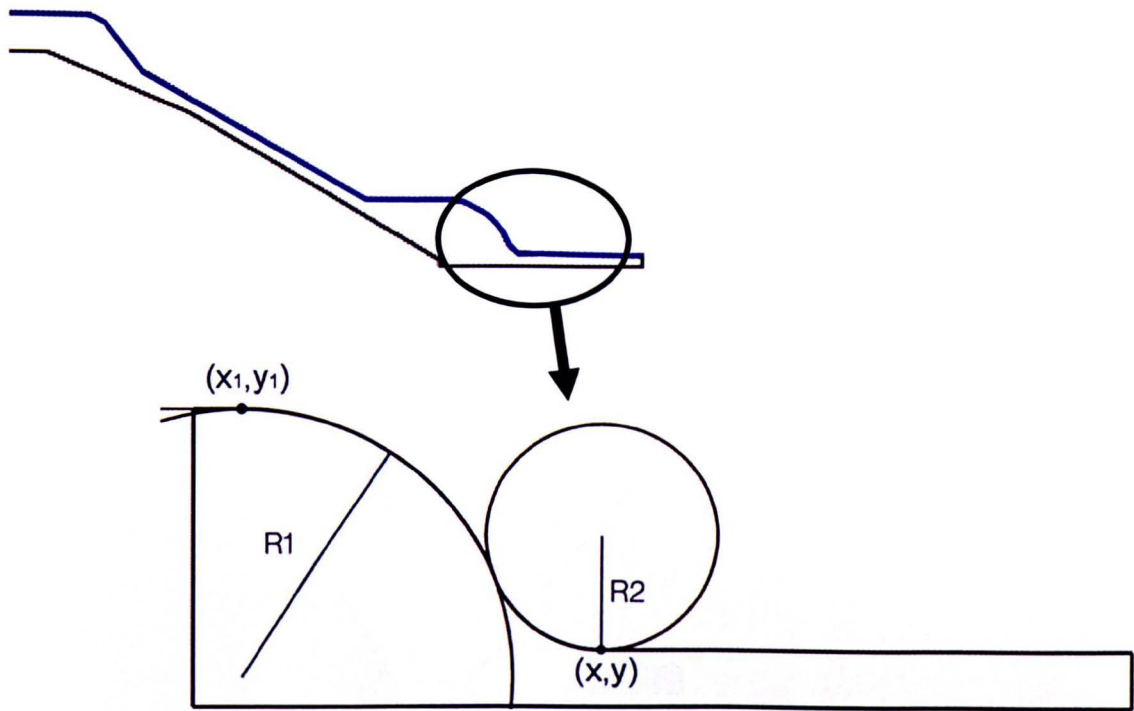
### 2-D Nozzle parameterisation

The aforementioned parameters affect the flow characteristics and especially the discharge coefficient. Apart from the physical constraints, the parameterisation of the geometry must also be well defined. Looking at the problem of parameterisation from the mathematical point of view, we must ensure that every set of parameters results in one and only one design.

In the case of the 2-D axisymmetric nozzle the parameters that are chosen to modify the geometry are the radius of curvature and the inlet hole diameter. The inlet hole diameter affects the conicity of the nozzle hole. Figure 3.2 represents the possible choices of parameters for modifying this geometry. Moreover, the radius  $R1$  changes the sac volume and as a result affects the overall performance of the nozzle. For this reason the radius  $R1$  is not included in the parameterisation scheme, and neither is the point  $(x1, y1)$ . On the other hand the radius of curvature  $R2$  at the hole inlet as well as the point  $(x, y)$  are used through the geometry modification.

The parameterisation scheme works in the following manner. Two parameters are used. The first one is a coefficient/number which is used to multiply the y-coordinate of the point  $(x, y)$ . This way we produce the modified hole wall that affects the hole entrance diameter and thus, the conicity of the nozzle hole. The second parameter is a radius producing a part of a circle, a.k.a arc, that replaces the sharp hole inlet of the initial nozzle. The arc is fitted tangential on the top part of the hole wall and the sac of the nozzle, the way it is shown in Figure 3.2. In this way, only the localised region affecting cavitation inception is parameterised and reproduced in every optimisation cycle. As it has already been mentioned, the parameters affected by this parameterisation are the hole inlet diameter and the radius of curvature and the conicity, while the hole length as well as the upstream nozzle geometry remain unaffected.

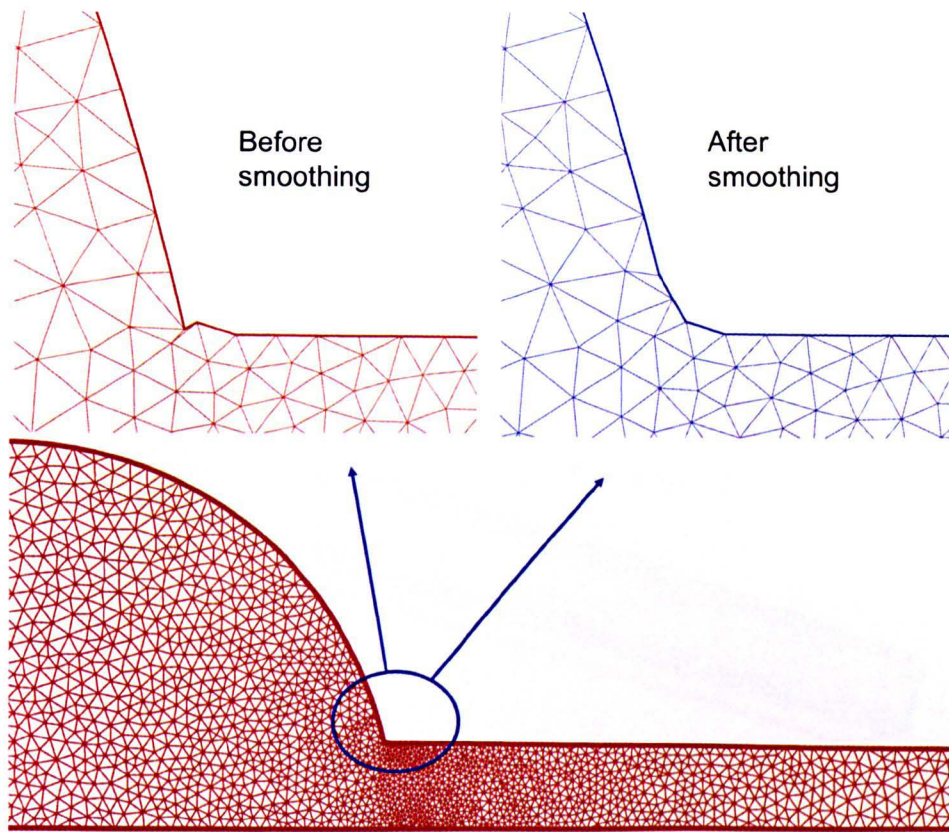




**Figure 3.2: Parameterisation used for 2-D axisymmetric single hole nozzle redesign**

The procedure we have described results to some complications in the produced geometry. Due to the inclination change of the hole wall, the fitted circle cannot always be tangential to both the hole wall and the sac. This results in discontinuity of the nozzle geometry and thus, bad quality produced grids. A bad quality grid may lead to non-converged solution of the flow solver or even the breakdown of the overall solution process. The necessity for applying smoothers appears; so a linear polynomial smoother is implemented in the parameterisation scheme. Figure 3.3 the effect of the smoother is shown on the geometry as well as on the produced mesh.



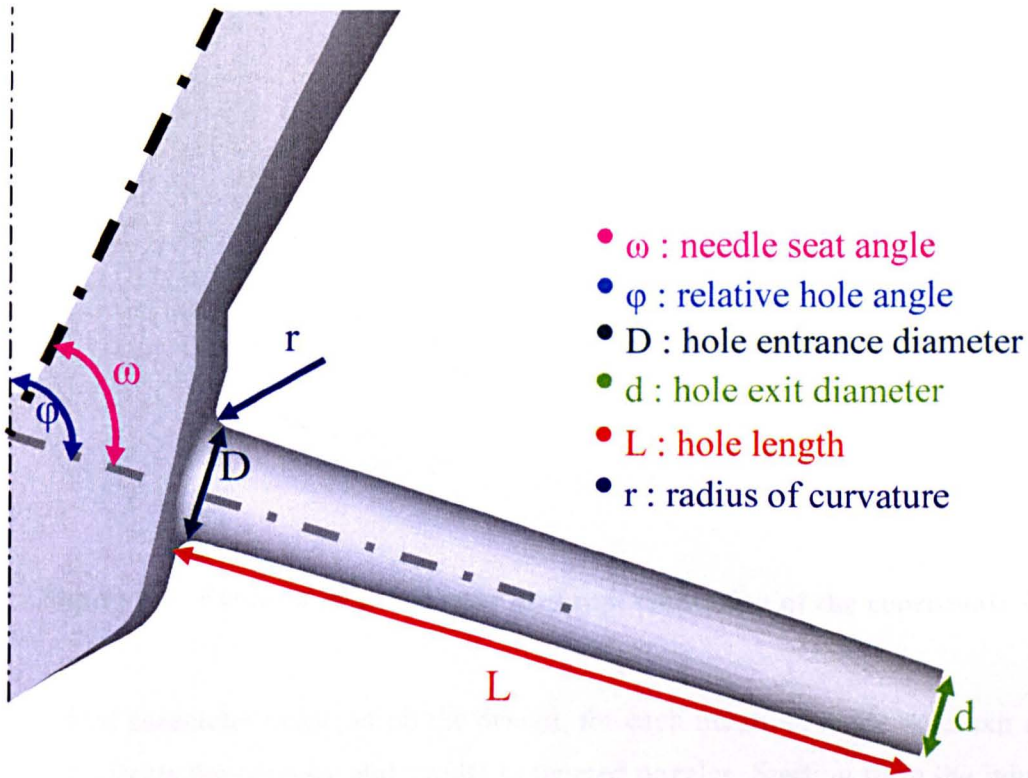


**Figure 3.3: Linear smoother for 2-D parameterisation**

### 3-D Nozzle parameterisation

Parameterising the whole 3-D nozzle is a complicated task. It is well known that, the CPU time required for the code to run is increasing with the number of cells. In this study we have used the symmetry attributes of the nozzle geometry and solved only half of the hole. This corresponds to the  $1/12^{\text{th}}$  of a typical six hole nozzle. Nevertheless, the parameterisation scheme developed in this study can easily be extended to the whole nozzle geometry.

A scheme similar to the 2-D case works also for the 3-D nozzle parameterisation. The parameters affecting the discharge coefficient in the 3-D case are shown in Figure 3.4. Again in this case the parameters used affect the radius of curvature at the hole inlet and the conicity; but also a new parameter for the needle seat angle has been introduced.

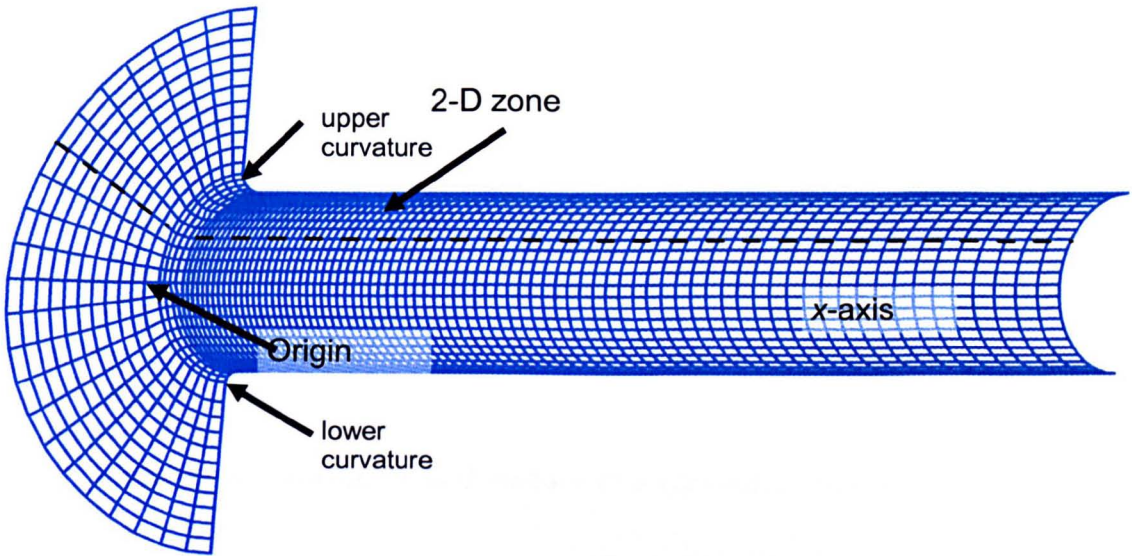


**Figure 3.4: Parameters that affect the discharge coefficient**

In order to be able to use attributes of the 2-D parameterisation scheme some manipulation of the nozzle topology is needed; a local coordinate system is defined to simplify the mathematical model used for the parameterisation.

The origin of the Cartesian system is located at the centre of the hole entrance, the x-axis is parallel to the hole wall (see Figure 3.4) and the xz-plane is the symmetry plane of the nozzle. This helps in the calculation of the inlet and exit diameters and the rotational matrices necessary for the parameterisation of the needle seat angle. The hole of the nozzle is divided in 'zones' along its length; the surface of the hole geometry is reduced into a set of 2-D lines parallel to the x-axis (Figure 3.5). This can be done easily provided that the surface mesh of the hole is structured or created by general quadrilateral shapes. A representation of the symmetric hole of a real nozzle, as it is meshed for the need of the parameterisation scheme, is shown in Figure 3.5. The 2-D zones are shown as well as the origin of the local coordinate system and the x-axis. So the problem is reduced to a 2-D parameterisation and treated almost the same way as the 2-D case, described in the previous paragraph.





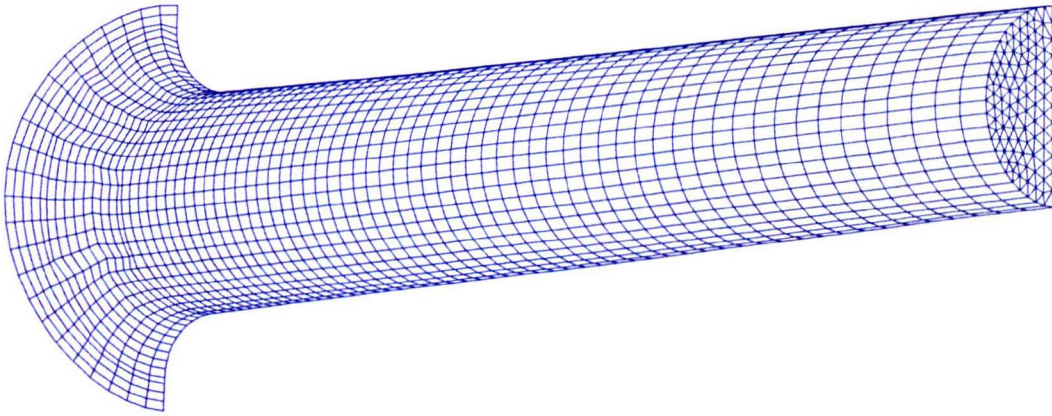
**Figure 3.5: Example of a 2-D zone and representation of the coordinate system**

The first parameter enforced on the design, for each iteration, is the hole exit diameter, which affects the conicity and results to tapered nozzles. Starting from the initial cylindrical hole geometry, a new hole exit is defined and the 2-D zones are inclined to produce the hole wall corresponding to this new exit hole. Then the needle seat angle is enforced by means of a rotational matrix turning all the surface points around the y-axis. The radius of curvature is treated in the same way as in the 2-D nozzle on every zone. In this case there is no need for smoothing since each zone of the body of the nozzle is linear in most cases. The conicity in the 3-D parameterisation is treated using the hole exit diameter instead of the hole entrance one. This is because in the 3-D case it is complicated to change the shape around of the hole entrance.

Using this parameterisation scheme, it is possible to define a different radius of curvature for each zone. For the purpose of producing more realistic nozzle shapes, it has been preferred, in this study, to parameterise only the radiuses of the first and the last zone. The in-between radiuses are calculated by interpolating the values of those two parameters. So the overall number of parameters in the 3-D case is reduced to four.

In Figure 3.6 the parameterised surface of a nozzle hole is presented. The hole exit surface mesh is produced during the parameterisation to aid the grid modification process, described in the next section. The new position of the mesh points is calculated using the ratio of the new over the initial exit diameter.





**Figure 3.6: Parameterised surface of a symmetric 3-D nozzle hole**

### 3.6 GRID GENERATION

The focus of the present and future work is on unstructured grid generation technologies, rather than on structured grids. The unstructured approach allows fully automated grid generation, with automatic grid error detection and refinement. Furthermore, the unstructured approach allows for virtually perfect load-balancing distributions for improved parallel efficiency. The parallelisation of the programme is in the future plans.

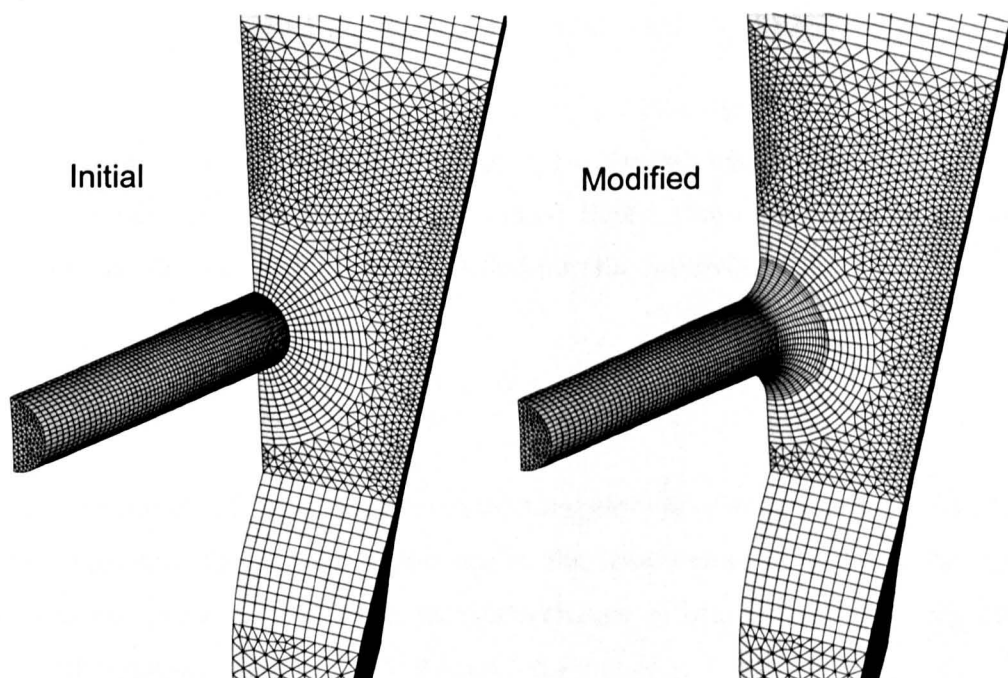
The scope of this study is not grid generation and only a brief presentation of the methods used in this study is presented. More details about various methods in grid generation can be found in books and conference proceedings devoted to this topic, e.g. Thompson et al [198] and Arcilla et al [199].

With regards to computational design optimisation, the unstructured mesh can be automatically regenerated within the solution process. This allows for automated shape modifications in addition to automated grid refinement. Through this process there is no need to recreate the data structure i.e.: define indexes, cell faces etc, because the necessary arrays are stored from the initial grid read.

For the 2-D cases the required computational unstructured meshes are generated using a typical advancing front method implemented in the computational code. The grid generation software is very fast, so it is repetitively used within each cycle to create the computational grid for every new nozzle.

For cases requiring more complicated meshes there is the option of modifying the initial unstructured mesh which is provided by a commercial programme. The grid modification method is based on a geometric progression movement of the mesh lines according to the new position of the modified surface geometry in every optimisation cycle. This method can be used in order to reduce the cost of the grid generation as it has already been stated. Also this method enables continuation of the flow equations solution from the previous optimisation cycle which results to extremely lower computational time. In some cases 90% reduction of the computational cost has been succeeded by using the solution of the previous flow field instead of starting the solution from the beginning.

The same method has been used for the 3-D cases where an automatic grid generator is not available. The initial mesh is provided using commercial software. As it has already been mentioned in the previous paragraph the hole of the nozzle is meshed using hexahedral cells for convenience in the parameterisation. The process of re-meshing is aided by moving the mesh of the hole exit from within the parameterisation scheme as it has been described in the previous section. The quality of the produced new grids is enforced by the use of smoothers for the parameterised geometry. Figure 3.7 shows an example of an initial and a modified nozzle mesh using the developed parameterisation and grid modification tools.



**Figure 3.7: Initial and modified nozzle mesh using the developed parameterisation and grid modification tools**

### 3.7 GOVERNING EQUATIONS FOR FLUID FLOWS

The Navier-Stokes equations have been used as constraints for the formulation of the optimisation problem. The two physical properties that are modelled by the flow equations are mass and linear momentum and these are presented and discussed in this section. The most basic conservation law in nature is that of mass. With the assumption of no sources such as chemical reactions or phase changes in the system, the equation of mass for incompressible flows reduces to:

$$\nabla \cdot \mathbf{u} = 0 \quad (3.8)$$

Newton's second law of motion states that the rate the change of momentum is equal to the external forces acting on the fluid, which for a viscous fluid, reads:

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = \nabla \cdot \mathbf{T} + \rho \mathbf{g} \quad (3.9)$$

In this study all fluids are assumed to be Newtonian which in form of equations the above assumption reads:

$$\mathbf{T} = -\left(p + \frac{2}{3}\mu \nabla \cdot \mathbf{u}\right) \mathbf{I} + \mu \left[ \nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T \right] \quad (3.10)$$

where  $\mu$  is the dynamic viscosity of the fluid,  $\mathbf{I}$  is the unit tensor,  $p$  is the static pressure. The Newtonian model applies to many actual fluids. Due to the incompressibility assumption, the stress tensor can be simplified into the following form:

$$\mathbf{T} = -p \mathbf{I} + \mu \left[ \nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T \right] \quad (3.11)$$

The energy conservation law states that the total amount of energy in an isolated system remains constant. In the case of this study, the temperature of the fluid has been assumed to be constant. In addition, no phase-change of heat transfer has been assumed. Thus, this equation has been omitted from the analysis.

We can summarise the steady Navier-Stokes equations for incompressible flows  $R(U)$ ,  $U = (\mathbf{u}, p)$ , which in tensorial form reads:

$$\begin{aligned}\nabla \cdot \mathbf{u} &= 0, \\ \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} - \mathbf{T}) &= \rho \mathbf{g},\end{aligned}\tag{3.12}$$

where,  $\mathbf{T} = -p\mathbf{I} + \mu \left[ \nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T \right]$ .

In this form the flow equations will be used in the next chapter for the derivation of the system of the adjoint equations.

### 3.7.1 Turbulence Modelling

Fluid flows in most engineering problems are turbulent. Nozzle flows are characterised by high Reynolds numbers and strong turbulent behaviour. The **GFS** code has several variations of the  $k - \varepsilon$  and the  $k - \omega$  turbulence models. The turbulence model is solved in an iterative manner together with the RANS equations. First, the Navier-Stokes equations are solved to compute the mean flow variables. Then, the turbulence model is solved to compute the turbulent eddy viscosity. For the purposes of this study the eddy viscosity is calculated by means of the *standard two-equation  $k - \varepsilon$*  model that was proposed by Launder and Spalding [200, 201]. This model features the solution of two additional scalar transport equations, namely one for  $k$ , which is the average turbulent kinetic energy (TKE), and one for  $\varepsilon$ , which is the dissipation rate of turbulent kinetic energy. Without getting into the theory of turbulence, the equations of the  $k - \varepsilon$  turbulence model are presented below.

So the transport equation of  $k$  is given, by:

$$\frac{\partial(\rho k)}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{u}} k) - \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right) = P - \rho \varepsilon \tag{3.13}$$

where  $P$  is the production of turbulent kinetic energy:

$$P = \mu_t (\nabla \otimes \bar{\mathbf{u}}) \cdot \left( (\nabla \otimes \bar{\mathbf{u}}) + (\nabla \otimes \bar{\mathbf{u}})^T \right) - \frac{2}{3} (\nabla \cdot \bar{\mathbf{u}}) (\rho k + \mu_t (\nabla \cdot \bar{\mathbf{u}})) \tag{3.14}$$

For incompressible flows  $\nabla \cdot \bar{\mathbf{u}} = 0$ , which simplifies into:

$$P = \mu_t (\nabla \otimes \bar{\mathbf{u}}) \cdot \left( (\nabla \otimes \bar{\mathbf{u}}) + (\nabla \otimes \bar{\mathbf{u}})^T \right) \tag{3.15}$$

After the transport equation of  $k$  the corresponding one for  $\varepsilon$  is presented:

$$\frac{\partial(\rho \varepsilon)}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{u}} \varepsilon) - \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right) = \frac{\varepsilon}{k} (C_{\varepsilon 1} P - C_{\varepsilon 2} \rho \varepsilon) + C_{\varepsilon 3} \rho \varepsilon \nabla \cdot \bar{\mathbf{u}} \tag{3.16}$$

Again for incompressible flows the last term in the above equation is omitted. Eddy viscosity is evaluated from the following equation:

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon} \quad (3.17)$$

In the above equations a number of constants have been introduced; their values are presented in Table 3.1.

Constant	$C_\mu$	$C_{\varepsilon 1}$	$C_{\varepsilon 2}$	$C_{\varepsilon 3}$	$\sigma_k$	$\sigma_\varepsilon$
Value	0.09	1.44	1.92	-0.33	1.0	1.3

**Table 3.1: Coefficients in the standard  $k$ - $\varepsilon$  model**

### 3.8 ADJOINT EQUATIONS ANALYSIS

As it has already been discussed the optimisation method used in this study is gradient-based or else called deterministic. This means that gradient information of the cost function is required to determine the search direction towards the optimum. This section is dedicated to the derivation of these gradients with respect to the design parameters. The continuous adjoint method has been used as the means to calculate these gradients. Extended analysis of the use of the method exists in this section. The adjoint system of equations is formulated along with the boundary conditions and the solution procedure resembling the flow solution is described. The difficulties of the implementation for different cost functions are discussed along with the way to overcome them. Finally the gradients are formulated for the different cost functions used in this study.

#### 3.8.1 Computing the gradients

Sensitivity derivatives provide information about the sensitivity of the flow with respect to changes in the design parameters. Traditionally, modifications are performed on a trial and error basis based on intuition. That is to say, if cavitation is occurring, one might say “let’s try rounding the tip a little more”. Sensitivity derivatives will help guide the designer to exactly where the rounding needs to be done and more importantly how much rounding is necessary to achieve the desired performance.

Differentiation of the flow equations using the adjoint method can be performed either in terms of the *continuous* or the *discrete* approach. For the continuous approach, the adjoint equations are extracted prior to discretisation. The analysis is performed by hand using the theory of Lagrange multipliers. For the discrete approach, the PDE’s are dif-



ferentiated after discretisation. In this work, the continuous approach to sensitivity analysis is used.

With gradient-based optimisation methods, the search direction is determined using the first derivatives of the objective and constraint functions with respect to the vector of independent design variables (i.e., sensitivity derivatives). This is not to say that the search direction is solely based on first-derivative information; it is possible to estimate higher-order derivatives using the computed first derivatives. A weakness of the adjoint approach is that there is no simple way to compute the Hessian matrix  $\partial^2 I_c / \partial D_i \partial D_j$  [56] even when the objective comes from a least-squares minimisation problem. Instead, the gradient-based optimisation method must construct an approximation to the Hessian matrix using information about the variation of the gradient at different points in the design space. In addition such methods usually determine a search direction and then find the optimum along the direction using line search. Both of these aspects results in more steps in the optimisation procedure than are required for the direct sensitivity approach, like finite differences, using approximation of the Hessian.

### 3.8.2 Formulation of adjoint equations

Efficient and accurate evaluation of the gradient of the cost function is one of the most important but difficult tasks in gradient-based optimisation. However, the adjoint-variable method offers an elegant mean for evaluating this gradient easily and accurately. Even in problems which are mathematically well posed, numerical resolution of the adjoint field can be exceedingly difficult to obtain, or the extraction of the gradient of the cost functional from the adjoint field exceedingly prone to amplification of numerical error, unless the proper care is taken in the definition of the adjoint field.

There is quite a bit of flexibility in how an adjoint-based optimisation problem is defined, and the choices made in this definition have an enormous impact on the rate of convergence of the resulting numerical algorithm [106]. The derivation of the adjoint-variable equations, as it is performed in the present study, is presented below.

### 3.8.3 Optimisation Procedure

For the analysis an arbitrary cost function is defined and noted  $I_C$ . The cost function  $I_C$  is a functional of the flow variables  $U$  and the vector of design variables  $D$ , defined on all or part of the boundary of the shape. In this way the procedure is generalised and can be used for any kind of cost function defined on the boundary of the shape which is modified.

In this study the optimisation process does not take into account the transient nature of the flow. Single-phase flow is sufficient to spot the cavitation inception area and intensity and measure them in each optimisation cycle. Moreover an optimisation procedure which takes into account the needle movement is really expensive and could be better dealt as a steady-state multi-point case. Consequently, the adjoint problem is solved as steady-state problem using the incompressible Navier-Stokes flow equations  $R(U)$  of eq. (3.12).

The system of flow equations  $R(U)$  is integrated and introduced as constraints to the optimisation problem. In this way it is ensured that the state variables are uniquely determined for a given set of parameters in the domain of interest. Soto and Löhnert [146, 147, 149] analysis has been followed for the extraction of the adjoint equations and their boundary conditions throughout this study. The cost function  $I_C$  is augmented to the weak form of the constraints, through the Lagrange multiplier  $\Psi$  to give the augmented cost function  $I_{aug}$ :

$$I_{aug} = I_C + \int_V \Psi \cdot R dV = I_C + \int_V \Psi_p \nabla \cdot \mathbf{u} dV + \int_V \Psi_u \cdot [\nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} - \mathbf{T})] dV \quad (3.18)$$

where,  $\Psi = (\Psi_u, \Psi_p)^T$ ,  $\Psi_u = (\Psi_1, \Psi_2, \Psi_3)$  is the vector of the costate variables and  $V$  is the computational domain. By application of the Green-Gauss theorem some surface integrals appear in the above equation (3.18) of the augmented cost function which takes its final form:

$$\begin{aligned} I_{aug} = I_C + \oint_{\partial V} \Psi_p \mathbf{u} \cdot \vec{n} ds + \oint_{\partial V} \Psi_u \cdot (\rho \mathbf{u} \otimes \mathbf{u} - \mathbf{T}) \cdot \vec{n} ds - \\ - \int_V \nabla \Psi_p \cdot \mathbf{u} dV + \int_V (\nabla \otimes \Psi_u) (\rho \mathbf{u} \otimes \mathbf{u} - \mathbf{T}) dV, \end{aligned} \quad (3.19)$$

where  $\partial V$  is the boundary around the domain  $V$  and  $\vec{n} = (n_x, n_y, n_z)$  is the outward normal vector along  $\partial V$ .

The augmented cost function  $I_{aug}$  is a functional of the flow variables  $U$ , the costate or adjoint variables  $\Psi$  and the vector of design variables  $D$ . An optimal point of the minimisation problem should meet the above necessary conditions, and is stated as:

$$(1) \quad R(U) = 0 \quad (3.20)$$

$$(2) \quad \frac{\partial I_{aug}}{\partial U} = 0 \quad (3.21)$$

$$(3) \quad \frac{\partial I_{aug}}{\partial D} = 0 \quad (3.22)$$

The first condition (3.20) is satisfied by the flow solution. The derivation of the adjoint equations results from the second condition (3.21). The design variables  $D$  are considered fixed while solving these equations from which the vector of costate variables  $\Psi$  is obtained. Having solved the adjoint equations and obtained the costate variables their values can be substituted to the third condition (3.22) giving the adjoint gradients with respect to the design variables  $D$ . So the design parameters  $D$  are updated using an optimisation method and the whole process is repeated. Details about the derivation of the adjoint equations using the second conditions and discussion about the gradients follow in the next paragraphs.

### 3.8.4 The adjoint equations

The basic idea of the adjoint approach is to define the adjoint problem so as to eliminate the explicit dependence of the cost function  $I_C$  on the flow variables  $U$  giving the adjoint form of the objective function. The derivation of the adjoint equations and the sensitivity derivatives of the cost function with respect to the design variables closely follow classical techniques from the calculus of variation and the theory of Lagrange multipliers. The sensitivity derivatives of  $I_C$  with respect to the design variables are obtained by taking the variational form of  $I_{aug}$ . The calculations in the present study are the work of the author. The extracted results are compared to the ones of Soto and Löhner [147]. For the derivation of the adjoint equations the second condition of eq. (3.21) is used. This condition can be expanded to give:

$$\frac{\partial I_{aug}}{\partial U} = \frac{\partial I_C}{\partial U} + \frac{\partial}{\partial U} \left( \int_V \Psi R dV \right) = 0 \quad (3.23)$$

Any modification/perturbation of the design variables results in a corresponding variation of the body shape. This, in return, produces a variation to the computational domain

and thus, the flow solution  $U$ . Having solved the flow equation and satisfied the first condition (3.20) the variation of the computational domain is introduced in the second condition by multiplying it with an arbitrary perturbation of the flow variables  $\delta U$  giving the variational form of the augmented cost function:

$$\delta I_{aug} = \frac{\partial I_{aug}}{\partial U} \delta U = \frac{\partial I_c}{\partial U} \delta U + \int_V \Psi \frac{\partial R}{\partial U} \delta U dV = 0 \quad (3.24)$$

The above procedure corresponds to an individual cycle of the optimisation process so the design variables are fixed during the derivation of the adjoint system of equations. A second order approximation of the variation of the flow equation reads:

$$\frac{\partial R}{\partial U} \delta U = \left\{ \begin{array}{l} \nabla \cdot \delta \mathbf{u} = 0 \\ \nabla \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) + \nabla \delta p - \nabla \cdot \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] = 0 \end{array} \right\} \quad (3.25)$$

where  $\delta \mathbf{u}$  and  $\delta p$  are perturbations of the velocity and pressure field respectively. The variation of the flow solution  $\frac{\partial R}{\partial U} \delta U$  is equal to zero because the variation of the source term  $\rho \mathbf{g}$  vanishes due to the assumed incompressibility of the fluid.

Introducing eq. (3.25) into (3.24) the expression for the variation of the augmented cost function  $\delta I_{aug}$  is obtained:

$$\begin{aligned} \delta I_{aug} &= \frac{\partial I_{aug}}{\partial U} \delta U = \frac{\partial I_c}{\partial U} \delta U + \int_V \Psi_p \nabla \cdot (\delta \mathbf{u}) dV + \\ &+ \int_V \Psi_u \nabla \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) + \Psi_u \nabla \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) + \Psi_u \nabla \delta p dV - \\ &- \int_V \Psi_u \nabla \cdot \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] dV \end{aligned} \quad (3.26)$$

Using the Green-Gauss theorem to integrate by parts in a way that the terms containing derivatives of  $\delta \mathbf{u}$  and  $\delta p$  are eliminated and splitting the resulting equation by taking the functional basis  $(\mathbf{u}, 0)$  and  $(0, p)$  to approximate  $\delta U$  the following expressions are obtained:

$$\begin{aligned}
0 = & \frac{\partial I_c}{\partial \mathbf{u}} \delta \mathbf{u} + \oint_{\partial V} \Psi_p \cdot \delta \mathbf{u} \bar{n} ds + \oint_{\partial V} \Psi_u \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) \bar{n} ds + \oint_{\partial V} \Psi_u \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) \bar{n} ds - \\
& - \oint_{\partial V} \Psi_u \cdot \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] \bar{n} ds \\
& - \int_V \nabla \Psi_p \cdot \delta \mathbf{u} dV - \int_V (\nabla \otimes \Psi_u) : (\rho \delta \mathbf{u} \otimes \mathbf{u}) dV - \int_V (\nabla \otimes \Psi_u) : (\rho \mathbf{u} \otimes \delta \mathbf{u}) dV + \\
& + \int_V (\nabla \otimes \Psi_u) : \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] dV, \quad (3.27)
\end{aligned}$$

$$0 = \frac{\partial I_c}{\partial p} \delta p + \oint_{\partial V} \Psi_u \cdot \delta p \bar{n} ds - \int_V \nabla \cdot \Psi_u \delta p dV$$

In order to obtain  $\delta I_{aug}$  independently from the variation of  $\delta U$  the corresponding integrals must vanish. If the cost function  $I_c$  is defined only on the boundary  $\partial V$  (or part of it) and if one takes into account that these equations have to be fulfilled  $\forall \delta \mathbf{u}$  and  $\forall \delta p$  then equations (3.27) can be reduced to a corresponding PDE problem using the following manipulation. Eliminating the field and surface integrals that multiplies the variation of the flow variables  $\delta U$  provide the costate equations along with their boundary conditions. In this respect those integrals are manipulated one by one in order to cancel their dependence on  $\delta \mathbf{u}$  and  $\delta p$ .

The second equation, of the system of equations (3.27), is dealt first due to its simplicity. By considering  $\delta p$  arbitrary only one field term is extracted straightforwardly which gives:

$$\int_V \nabla \cdot \Psi_u \delta p dV \Rightarrow \nabla \cdot \Psi_u = 0 \quad (3.28)$$

The above equation is similar and corresponds to the continuity equation of the flow system.

From the first equation, of the system of equations (3.27), the term multiplying  $\Psi_p$  can be reduced to give:

$$-\int_V \nabla \Psi_p \cdot \delta \mathbf{u} dV \Rightarrow -\nabla \Psi_p \quad (3.29)$$

which corresponds to the pressure term of the momentum equation. For the rest of the terms some calculations are needed to derive the PDEs. Also the conservation property is not straight forward for the convection terms.

The adjoint system of equations actually resembles the flow system in the sense that the convection and diffusion terms are easily identifiable. The most important difference that results from the following analysis is that the adjoint system is linear. For the derivation of the remaining terms the first the assumption of a Cartesian reference frame is made and generalisation follows.

### Convection terms

The convection terms of the adjoint equations are derived from the term  $\int_V \Psi_u \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV$  of the augmented cost function  $I_{aug}$ . The variation of the above term reads:

$$\begin{aligned} \int_V \Psi_u \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV &\Rightarrow \oint_{\partial V} \Psi_u \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) \vec{n} ds + \oint_{\partial V} \Psi_u \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) \vec{n} ds - \\ &- \int_V (\nabla \otimes \Psi_u) : (\rho \delta \mathbf{u} \otimes \mathbf{u}) dV - \int_V (\nabla \otimes \Psi_u) : (\rho \mathbf{u} \otimes \delta \mathbf{u}) dV \end{aligned} \quad (3.30)$$

<sup>1</sup> For the system of equations (3.30) the field integrals are used for the adjoint equations while the boundary integrals along with the cost function  $I_c$  provide the boundary conditions. The terms are analysed one by one to eliminate the dependence of  $\delta \mathbf{u}$ . If the vector of costate variables in the 3-D space is  $\Psi_u = (\Psi_1, \Psi_2, \Psi_3)$  and the vector of curvilinear coordinates is  $\mathbf{X} = (x, y, z)$ , the terms included in the first field integral of equation (3.30) can be calculated to give:

---

<sup>1</sup> The operators used in equation (3.30) have the following formulas if  $A = (a_1, a_2, a_3)$  and  $B = (b_1, b_2, b_3)$  :

$$A \otimes B = \begin{pmatrix} a_1 b_1 & a_2 b_1 & a_3 b_1 \\ a_1 b_2 & a_2 b_2 & a_3 b_2 \\ a_1 b_3 & a_2 b_3 & a_3 b_3 \end{pmatrix} \text{ and } A : B = (a_1 b_1 + a_1 b_2 + a_1 b_3 + a_2 b_1 + a_2 b_2 + a_2 b_3 + \\ + a_3 b_1 + a_3 b_2 + a_3 b_3)$$

$$(\nabla \otimes \Psi_{\mathbf{u}}):(\rho \delta \mathbf{u} \otimes \mathbf{u}) = \begin{pmatrix} \frac{\partial \Psi_1}{\partial x} & \frac{\partial \Psi_2}{\partial x} & \frac{\partial \Psi_3}{\partial x} \\ \frac{\partial \Psi_1}{\partial y} & \frac{\partial \Psi_2}{\partial y} & \frac{\partial \Psi_3}{\partial y} \\ \frac{\partial \Psi_1}{\partial z} & \frac{\partial \Psi_2}{\partial z} & \frac{\partial \Psi_3}{\partial z} \end{pmatrix} : \begin{pmatrix} \delta(\rho u)u & \delta(\rho v)u & \delta(\rho w)u \\ \delta(\rho u)v & \delta(\rho v)v & \delta(\rho w)v \\ \delta(\rho u)w & \delta(\rho v)w & \delta(\rho w)w \end{pmatrix} \quad (3.31)$$

while  $\rho$  can be excluded from the variation since incompressible flow has been assumed, so the above expression expands to:

$$\begin{aligned} (\nabla \otimes \Psi_{\mathbf{u}}):(\rho \delta \mathbf{u} \otimes \mathbf{u}) = & \rho \left( \frac{\partial \Psi_1}{\partial x}(\delta u)u + \frac{\partial \Psi_2}{\partial x}(\delta u)v + \frac{\partial \Psi_3}{\partial x}(\delta u)w + \right. \\ & + \frac{\partial \Psi_1}{\partial y}(\delta v)u + \frac{\partial \Psi_2}{\partial y}(\delta v)v + \frac{\partial \Psi_3}{\partial y}(\delta v)w + \\ & \left. + \frac{\partial \Psi_1}{\partial z}(\delta w)u + \frac{\partial \Psi_2}{\partial z}(\delta w)v + \frac{\partial \Psi_3}{\partial z}(\delta w)w \right) \end{aligned} \quad (3.32)$$

This, by eliminating the dependence of  $\delta \mathbf{u}$  from equation (3.32) gives:

$$(\nabla \otimes \Psi_{\mathbf{u}}):(\rho \delta \mathbf{u} \otimes \mathbf{u}) = \rho \delta \mathbf{u} \cdot \begin{pmatrix} u \frac{\partial \Psi_1}{\partial x} + v \frac{\partial \Psi_2}{\partial x} + w \frac{\partial \Psi_3}{\partial x} \\ u \frac{\partial \Psi_1}{\partial y} + v \frac{\partial \Psi_2}{\partial y} + w \frac{\partial \Psi_3}{\partial y} \\ u \frac{\partial \Psi_1}{\partial z} + v \frac{\partial \Psi_2}{\partial z} + w \frac{\partial \Psi_3}{\partial z} \end{pmatrix} \quad (3.33)$$

The above expression can be generalised for arbitrary cell shape in tensorial form and reads:

$$(\nabla \otimes \Psi_{\mathbf{u}}):(\rho \delta \mathbf{u} \otimes \mathbf{u}) = \delta \mathbf{u} \cdot (\nabla \otimes \Psi_{\mathbf{u}}) \rho \mathbf{u} \quad (3.34)$$

Following the same manipulation the terms of the second field integral of equation (3.30), gives:

$$\begin{aligned} (\nabla \otimes \Psi_{\mathbf{u}}):(\rho \mathbf{u} \otimes \delta \mathbf{u}) = & \rho \left( \frac{\partial \Psi_1}{\partial x}(\delta u)u + \frac{\partial \Psi_2}{\partial x}(\delta v)u + \frac{\partial \Psi_3}{\partial x}(\delta w)u + \right. \\ & + \frac{\partial \Psi_1}{\partial y}(\delta u)v + \frac{\partial \Psi_2}{\partial y}(\delta v)v + \frac{\partial \Psi_3}{\partial y}(\delta w)v + \\ & \left. + \frac{\partial \Psi_1}{\partial z}(\delta u)w + \frac{\partial \Psi_2}{\partial z}(\delta v)w + \frac{\partial \Psi_3}{\partial z}(\delta w)w \right) \end{aligned} \quad (3.35)$$

which, by eliminating the dependence of  $\delta \mathbf{u}$  and writing the expression (3.35) in a generalised tensorial form, reads:

$$(\nabla \otimes \Psi_{\mathbf{u}}):(\rho \mathbf{u} \otimes \delta \mathbf{u}) = \delta \mathbf{u} \cdot (\rho \mathbf{u} \nabla \otimes \Psi_{\mathbf{u}}) \quad (3.36)$$

The terms multiplying  $\delta \mathbf{u}$  in equations (3.34) and (3.36) can be further analysed to give an orthogonal and a non-orthogonal contribution, which reads:

$$\begin{aligned}
 (\nabla \otimes \Psi_{\mathbf{u}}) \rho \mathbf{u} + \rho \mathbf{u} \nabla \otimes \Psi_{\mathbf{u}} &= \\
 &= \begin{pmatrix} \frac{\partial \Psi_1}{\partial x} & \frac{\partial \Psi_2}{\partial x} & \frac{\partial \Psi_3}{\partial x} \\ \frac{\partial \Psi_1}{\partial y} & \frac{\partial \Psi_2}{\partial y} & \frac{\partial \Psi_3}{\partial y} \\ \frac{\partial \Psi_1}{\partial z} & \frac{\partial \Psi_2}{\partial z} & \frac{\partial \Psi_3}{\partial z} \end{pmatrix} \begin{pmatrix} \rho u \\ \rho v \\ \rho w \end{pmatrix} + (\rho u \quad \rho v \quad \rho w) \begin{pmatrix} \frac{\partial \Psi_1}{\partial x} & \frac{\partial \Psi_2}{\partial x} & \frac{\partial \Psi_3}{\partial x} \\ \frac{\partial \Psi_1}{\partial y} & \frac{\partial \Psi_2}{\partial y} & \frac{\partial \Psi_3}{\partial y} \\ \frac{\partial \Psi_1}{\partial z} & \frac{\partial \Psi_2}{\partial z} & \frac{\partial \Psi_3}{\partial z} \end{pmatrix} = \\
 &= \rho \left( \frac{\partial \Psi_1}{\partial x} u + \frac{\partial \Psi_2}{\partial x} v + \frac{\partial \Psi_3}{\partial x} w, \frac{\partial \Psi_1}{\partial y} u + \frac{\partial \Psi_2}{\partial y} v + \frac{\partial \Psi_3}{\partial y} w, \frac{\partial \Psi_1}{\partial z} u + \frac{\partial \Psi_2}{\partial z} v + \frac{\partial \Psi_3}{\partial z} w \right) + \\
 &+ \rho \left( \frac{\partial \Psi_1}{\partial x} u + \frac{\partial \Psi_1}{\partial y} v + \frac{\partial \Psi_1}{\partial z} w, \frac{\partial \Psi_2}{\partial x} u + \frac{\partial \Psi_2}{\partial y} v + \frac{\partial \Psi_2}{\partial z} w, \frac{\partial \Psi_3}{\partial x} u + \frac{\partial \Psi_3}{\partial y} v + \frac{\partial \Psi_3}{\partial z} w \right)
 \end{aligned} \tag{3.37}$$

The first term is non-orthogonal and cannot be analysed any further so this term contributes to the r.h.s of the equation in the form:

$$(\nabla \otimes \Psi_{\mathbf{u}}) \rho \mathbf{u} = \rho \mathbf{u} (\nabla \otimes \Psi_{\mathbf{u}})^T \tag{3.38}$$

The second term of equation (3.37) can be further analysed considering the fact that the velocity field doesn't change during the solution of the adjoint field. In this way we can include the velocity components into the partial derivatives of the costate variables in the following manner:

$$\begin{aligned}
 \rho \mathbf{u} \nabla \otimes \Psi_{\mathbf{u}} &= \\
 &= \rho \left( \frac{\partial u \Psi_1}{\partial x} + \frac{\partial v \Psi_1}{\partial y} + \frac{\partial w \Psi_1}{\partial z}, \frac{\partial u \Psi_2}{\partial x} + \frac{\partial v \Psi_2}{\partial y} + \frac{\partial w \Psi_2}{\partial z}, \frac{\partial u \Psi_3}{\partial x} + \frac{\partial v \Psi_3}{\partial y} + \frac{\partial w \Psi_3}{\partial z} \right) = \\
 &= \nabla \cdot (\rho \mathbf{u} \otimes \Psi_{\mathbf{u}})
 \end{aligned} \tag{3.39}$$

The above analysis is necessary to ensure that the terms are conservative so the FV method is applicable. Also this way a convection term appears which is very similar to the flow equations convection term, with the only difference that this term is linear so there is no need for special treatment as in the case of the velocity flux in section 4.5.3.

Concluding the analysis for the convection part of the adjoint equations the following convection terms for the costate equations are extracted:



$$\begin{aligned}
& -\int_V (\nabla \otimes \Psi_u) : (\rho \delta \mathbf{u} \otimes \mathbf{u}) dV - \int_V (\nabla \otimes \Psi_u) : (\rho \mathbf{u} \otimes \delta \mathbf{u}) dV \Rightarrow \\
& -\nabla \cdot (\rho \mathbf{u} \otimes \Psi_u) - \rho \mathbf{u} (\nabla \otimes \Psi_u)^T
\end{aligned} \tag{3.40}$$

In the next paragraph the diffusion terms are analysed.

### Diffusion terms

The derivation of the diffusion terms of the adjoint equation is not as obvious as the derivation convection terms. Due to the presence of the derivatives of the velocity components  $u$ ,  $v$  and  $w$  appearing in the diffusion part of the flow equations (3.12) the grouping of terms multiplying the variation of velocities  $\delta u$ ,  $\delta v$  and  $\delta w$  is not straightforward. The elimination of the dependence of the variation  $\delta U$  is actually accomplished by a second application of the Green-Gauss theorem. This gives rise to second derivatives of the co-state variables multiplying the variations  $\delta U$  and surface integrals containing first derivatives of the vector of the costate variables  $\Psi$ .

To proceed with the analysis as in the previous paragraph we consider the term

$$-\int_V \Psi_u \nabla \cdot \left[ \mu (\nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T) \right] dV \text{ of the augmented cost function } I_{aug} \text{ (3.26) that}$$

contributes to the diffusion part of the adjoint equations. The variation of the above term reads:

$$\begin{aligned}
& -\int_V \Psi_u \nabla \cdot \left[ \mu (\nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T) \right] dV \stackrel{\delta}{\Rightarrow} -\oint_{\partial V} \Psi_u \cdot \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] \tilde{\mathbf{n}} ds + \\
& + \int_V (\nabla \otimes \Psi_u) : \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] dV
\end{aligned} \tag{3.41}$$

Following the procedure of the previous paragraph the above term can be expanded to give:

$$\begin{aligned}
& (\nabla \otimes \Psi_u) : \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] = \\
& = \begin{pmatrix} \frac{\partial \Psi_1}{\partial x} & \frac{\partial \Psi_2}{\partial x} & \frac{\partial \Psi_3}{\partial x} \\ \frac{\partial \Psi_1}{\partial y} & \frac{\partial \Psi_2}{\partial y} & \frac{\partial \Psi_3}{\partial y} \\ \frac{\partial \Psi_1}{\partial z} & \frac{\partial \Psi_2}{\partial z} & \frac{\partial \Psi_3}{\partial z} \end{pmatrix} : \mu \left[ \begin{pmatrix} \frac{\partial(\delta u)}{\partial x} & \frac{\partial(\delta v)}{\partial x} & \frac{\partial(\delta w)}{\partial x} \\ \frac{\partial(\delta u)}{\partial y} & \frac{\partial(\delta v)}{\partial y} & \frac{\partial(\delta w)}{\partial y} \\ \frac{\partial(\delta u)}{\partial z} & \frac{\partial(\delta v)}{\partial z} & \frac{\partial(\delta w)}{\partial z} \end{pmatrix} + \begin{pmatrix} \frac{\partial(\delta u)}{\partial x} & \frac{\partial(\delta u)}{\partial y} & \frac{\partial(\delta u)}{\partial z} \\ \frac{\partial(\delta v)}{\partial x} & \frac{\partial(\delta v)}{\partial y} & \frac{\partial(\delta v)}{\partial z} \\ \frac{\partial(\delta w)}{\partial x} & \frac{\partial(\delta w)}{\partial y} & \frac{\partial(\delta w)}{\partial z} \end{pmatrix} \right] = \\
& = \begin{pmatrix} \frac{\partial \Psi_1}{\partial x} & \frac{\partial \Psi_2}{\partial x} & \frac{\partial \Psi_3}{\partial x} \\ \frac{\partial \Psi_1}{\partial y} & \frac{\partial \Psi_2}{\partial y} & \frac{\partial \Psi_3}{\partial y} \\ \frac{\partial \Psi_1}{\partial z} & \frac{\partial \Psi_2}{\partial z} & \frac{\partial \Psi_3}{\partial z} \end{pmatrix} : \mu \begin{pmatrix} 2 \frac{\partial(\delta u)}{\partial x} & \frac{\partial(\delta v)}{\partial x} + \frac{\partial(\delta u)}{\partial y} & \frac{\partial(\delta w)}{\partial x} + \frac{\partial(\delta u)}{\partial z} \\ \frac{\partial(\delta u)}{\partial y} + \frac{\partial(\delta v)}{\partial x} & 2 \frac{\partial(\delta v)}{\partial y} & \frac{\partial(\delta w)}{\partial y} + \frac{\partial(\delta v)}{\partial z} \\ \frac{\partial(\delta u)}{\partial z} + \frac{\partial(\delta w)}{\partial x} & \frac{\partial(\delta v)}{\partial z} + \frac{\partial(\delta w)}{\partial y} & 2 \frac{\partial(\delta w)}{\partial z} \end{pmatrix} \quad (3.42)
\end{aligned}$$

where Cartesian reference frame has again been assumed. At this point another assumption is necessary. In the derivation of the viscous adjoint terms, the variation of the turbulent viscosity  $\delta\mu$  is considered negligible and does not play significant part in the gradient calculation. The perturbation of turbulent viscosity is considered negligible, through each geometry modification. This is a convenient assumption from the numerical point of view and quite realistic considering the fact that the geometry changes slightly in every optimisation cycle. In the case of considering turbulent perturbation another two adjoint equations for the turbulence model will appear. In the case of turbulent flow, if the flow variations are found to result in significant changes in the turbulent viscosity, it may eventually be necessary to include its variation in the calculations [130]. This complication is not handled in the present study but the reader can refer to Anderson's work [43] for more details on how to derive the adjoint equations for the turbulent viscosity  $\mu$ .

Having these assumptions in mind the turbulent viscosity  $\mu$  is excluded from the gradients and variations and the above expression expands to:

$$\begin{aligned}
& (\nabla \otimes \Psi_{\mathbf{u}}) : \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] = \\
& = \mu \left[ \frac{\partial \Psi_1}{\partial x} 2 \frac{\partial (\delta u)}{\partial x} + \left( \frac{\partial \Psi_2}{\partial x} + \frac{\partial \Psi_1}{\partial y} \right) \left( \frac{\partial (\delta u)}{\partial y} + \frac{\partial (\delta v)}{\partial x} \right) + \right. \\
& \quad + \frac{\partial \Psi_2}{\partial y} 2 \frac{\partial (\delta v)}{\partial y} + \left( \frac{\partial \Psi_3}{\partial x} + \frac{\partial \Psi_1}{\partial z} \right) \left( \frac{\partial (\delta u)}{\partial z} + \frac{\partial (\delta w)}{\partial x} \right) \\
& \quad \left. + \frac{\partial \Psi_3}{\partial z} 2 \frac{\partial (\delta w)}{\partial z} + \left( \frac{\partial \Psi_3}{\partial y} + \frac{\partial \Psi_2}{\partial z} \right) \left( \frac{\partial (\delta v)}{\partial z} + \frac{\partial (\delta w)}{\partial y} \right) \right] \quad (3.43)
\end{aligned}$$

At this point a second application of the Green-Gauss theorem takes place. As it has been mentioned before this is a necessary step for the elimination of the variation of the flow variables  $\delta U$  due to the presence of the partial derivatives of the variation of velocity components  $\delta u$ ,  $\delta v$  and  $\delta w$  appearing in the above expression. The application of integration by part gives in equation (3.43) the following form:

$$\begin{aligned}
& \int_V (\nabla \otimes \Psi_{\mathbf{u}}) : \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] dV = \\
& = -\mu \int_V \left[ \frac{\partial}{\partial x} \left( \frac{\partial \Psi_1}{\partial x} \right) 2 \delta u + \frac{\partial}{\partial y} \left( \frac{\partial \Psi_2}{\partial x} + \frac{\partial \Psi_1}{\partial y} \right) \delta u + \frac{\partial}{\partial x} \left( \frac{\partial \Psi_2}{\partial x} + \frac{\partial \Psi_1}{\partial y} \right) \delta v + \right. \\
& \quad + \frac{\partial}{\partial y} \left( \frac{\partial \Psi_2}{\partial y} \right) 2 \delta v + \frac{\partial}{\partial z} \left( \frac{\partial \Psi_3}{\partial x} + \frac{\partial \Psi_1}{\partial z} \right) \delta u + \frac{\partial}{\partial x} \left( \frac{\partial \Psi_3}{\partial x} + \frac{\partial \Psi_1}{\partial z} \right) \delta w + \\
& \quad + \frac{\partial}{\partial z} \left( \frac{\partial \Psi_3}{\partial z} \right) 2 \delta w + \frac{\partial}{\partial z} \left( \frac{\partial \Psi_3}{\partial y} + \frac{\partial \Psi_2}{\partial z} \right) \delta v + \frac{\partial}{\partial y} \left( \frac{\partial \Psi_3}{\partial y} + \frac{\partial \Psi_2}{\partial z} \right) \delta w \Big] dV + \\
& \quad + \mu \oint_{\partial V} \bar{n} \left[ 2 (\nabla \otimes \Psi_{\mathbf{u}}) + (\nabla \otimes \Psi_{\mathbf{u}})^T \right] \delta \mathbf{u} ds \quad (3.44)
\end{aligned}$$

The boundary integral contributes to the sensitivity derivatives while the field integral is written in such a way that the dependence of  $\delta \mathbf{u}$  can now be easily eliminated. Writing the expression (3.44) in a generalised tensorial form we conclude that the diffusion term of the system of adjoint equations, reads:

$$(\nabla \otimes \Psi_{\mathbf{u}}) : \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] = -\delta \mathbf{u} \cdot \nabla \cdot \left[ \mu \left( \nabla \otimes \Psi_{\mathbf{u}} + (\nabla \otimes \Psi_{\mathbf{u}})^T \right) \right] + SD \quad (3.45)$$

In this case again we can split the diffusion term (3.45) into an orthogonal and a non-orthogonal part the same way that is done for the flow equations:

$$-\nabla \cdot \left[ \mu \left( \nabla \otimes \Psi_u + (\nabla \otimes \Psi_u)^T \right) \right] = -\nabla \cdot \left[ \mu (\nabla \otimes \Psi_u) \right] - (\nabla \otimes \Psi_u) \cdot \nabla \mu \quad (3.46)$$

The first term goes to the l.h.s of the adjoint equations and the second to the r.h.s. In the next paragraph the adjoint system of equations is summarized.

### 3.8.5 Final form of the adjoint system of equations

In this paragraph the contribution of the terms (3.28), (3.29), (3.40) and (3.46), derived from the previous analysis, is summarised and the adjoint system of equations is formed. For completeness time variation and compressibility will be included although all the problems solved in this study are steady state and incompressible. After the addition of the time variation of the costate variables, the adjoint equations take the form:

$$\begin{aligned} \frac{\partial \rho}{\partial t} - \nabla \cdot \Psi_u &= 0 \\ \frac{\partial}{\partial t} (\rho \Psi_u) - \rho u \left( \nabla \otimes \Psi_u + (\nabla \otimes \Psi_u)^T \right) - \nabla \cdot \mathbf{T}_{adj} &= 0, \end{aligned} \quad (3.47)$$

where  $\mathbf{T}_{adj} = -\rho \Psi_p I + \mu \left( \nabla \otimes \Psi_u + (\nabla \otimes \Psi_u)^T \right)$ .

The above equations (3.47) are very similar to the Navier-Stokes and can be characterised as the costate continuity and momentum equations without of course having the corresponding physical meaning. Considering the analysis of the previous paragraphs and the assumptions made the above system can be actually transformed to give:

$$\begin{aligned} \nabla \cdot \Psi_u &= 0 \\ -\nabla \cdot (\rho u \otimes \Psi_u) - \nabla \cdot \left[ \mu (\nabla \otimes \Psi_u) \right] &= \\ = \nabla \Psi_p + (\nabla \otimes \Psi_u) \cdot \nabla \mu + \rho u (\nabla \otimes \Psi_u)^T \end{aligned} \quad (3.48)$$

One important detail in the above re-arrangement is the separation of the diffusion term into orthogonal and non-orthogonal contributions; the orthogonal term, as it was discussed in section 3.8.4, can be linearised and it is located on the l.h.s. of the equation, whereas the non-orthogonal contribution has been moved to the r.h.s. of the equation. Although obvious, it should be mentioned that all terms on the l.h.s. of equation are linear functions of the unknown  $\Psi$ . The r.h.s. of the above equation contains all the terms which could not be linearised.

The resulting system of equations (3.48) is a convection-diffusion system, where information transfer takes place in the opposite direction of the information propagation of the costate variables in relation to the flow variables.

The remaining surface integral multiplying variations along boundaries, contribute to the boundary conditions for the adjoint equations. In those integrals, the terms arising from the second integration by parts of the diffusion part which contribute to the sensitivity derivatives, are not included, as it has been already mentioned. Summarising the remaining boundary terms we conclude the following expression for the boundary conditions of the adjoint equations:

$$\begin{aligned}
 0 &= \frac{\partial I_C}{\partial \mathbf{u}} \delta \mathbf{u} + \oint_{\partial V} \Psi_p \cdot \rho \delta \mathbf{u} \bar{n} ds + \oint_{\partial V} \Psi_u \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) \bar{n} ds + \\
 &+ \oint_{\partial V} \Psi_u \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) \bar{n} ds - \oint_{\partial V} \Psi_u \cdot \left[ \mu \left( \nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T \right) \right] \bar{n} ds \quad (3.49) \\
 0 &= \frac{\partial I_C}{\partial p} \delta p + \oint_{\partial V} \Psi_p \cdot \delta p \bar{n} ds
 \end{aligned}$$

It is obvious from the above expression (3.49) that the boundary conditions can only be derived after the definition of the cost function  $I_C$ . The same applies also for the sensitivity derivatives which are actually the gradients of the augmented cost  $I_{aug}$  function with respect to the design variables and are the result of the third condition (3.22).

### 3.9 BOUNDARY CONDITIONS

The choice of the Navier-Stokes equations as constraints in the optimisation problem provides the boundary terms of equation (3.49) independently of the choice of cost function. Of course for the extraction of the boundary conditions the cost function must be specified so that the calculation of its gradients, appearing in equation (3.49), is possible. This section comprises a detailed discussion about the derivation of the boundary conditions for the adjoint equations according to the two different cost functions described in section 3.4. The choice of these cost functions is such to fulfil the needs of the optimisation problem but also to provide challenges in the adjoint analysis. In the next paragraphs the two cases are discussed separately.

### 3.9.1 Target pressure distribution

The most typical inverse design optimisation problem in CFD aims at finding the shape of an engine or aerodynamic component such that the resulting pressure distribution along the shapes surface will minimise the least squares distance from a predescribed pressure distribution. This is expressed in the form of a cost function as follows:

$$I_c = \frac{1}{2} \oint_W (p - \bar{p}_t)^2 ds \quad (3.50)$$

where  $p$  and  $\bar{p}_t$  represent the current and target pressure respectively.  $W$  stands for wall and is the boundary of the geometry to be modified. The square power in the cost function ensures positive Hessian and thus convexity and it is used for mathematical reasons. The above cost function cannot be used in the case of the Navier-Stokes equations used as constraints and in the literature this function is considered as inadmissible [114, 128]. There is a lot of published research about the topic of admissible and inadmissible cost functions. Arian and Ta'asan in [116] define these problems and propose a solution by including some extra terms in the Langragian cost functional. These terms are obtained from the restriction of the interior PDE to the control surface. Demonstration of the explicit derivation of the adjoint equations for “inadmissible” cost functionals were given for the potential, Euler and Navier-Stokes equations.

Generally, suitable cost functions are composed of terms that will appropriately balance the boundary terms from the residuals. This means that for a well-posed adjoint problem, there is a restriction for the cost function  $I_c$ . The definition of  $I_c$  must include terms which satisfy equation (3.49). Restriction of the same nature was recognised in [7]. Returning to the case of equation (3.50) it is not immediately obvious that the specification of a pressure distribution is allowable because of the absence of viscous terms in the cost function. The cost function is enforced in the wall boundary around the sharp hole inlet. Since the Navier-Stokes equations have been introduced as constraints in the optimisation problem the no-slip condition applies in the wall boundaries. Applying the no-slip flow condition on equation (3.49) we obtain:

$$\begin{aligned} \frac{\partial I_c}{\partial \mathbf{u}} \delta \mathbf{u} &= \oint_W \Psi_{\mathbf{u}} \cdot \left[ \mu (\nabla \otimes \delta \mathbf{u} + (\nabla \otimes \delta \mathbf{u})^T) \right] \bar{\mathbf{n}} ds \\ \frac{\partial I_c}{\partial p} \delta p &= - \oint_W \Psi_p \cdot \delta p \bar{\mathbf{n}} ds \end{aligned} \quad (3.51)$$

It is obvious that equation (3.50) could never satisfy the first equation in the above system. However, a suitable cost function can be obtained by including another term in the cost function to obtain the cost function of equation (3.3). Equation (3.3) satisfies the boundary conditions (3.52) ensuring a well-posed adjoint problem. One should be aware that by the combination of the continuity equation and the no-slip boundary conditions the introduction of the extra term in  $I_c$ , as suggested in section 3.4, does not modify the minimisation problem.

The variation of the cost function of equation (3.3) with respect to the flow variables takes the form:

$$\begin{aligned}\frac{\partial I_c}{\partial \mathbf{u}} \delta \mathbf{u} &= - \oint_w (p - \bar{p}_i) \mu \frac{\partial (\delta \mathbf{u})}{\partial \bar{n}} \cdot \bar{\mathbf{n}} ds \\ \frac{\partial I_c}{\partial p} \delta p &= \oint_w (p - \bar{p}_i) \delta p ds - \oint_w \mu \frac{\partial \mathbf{u}}{\partial \bar{n}} \cdot \bar{\mathbf{n}} \delta p ds\end{aligned}\tag{3.53}$$

Replacing these variations in equation (3.53) and taking into account the continuity equation that indicates  $\frac{\partial \mathbf{u}}{\partial \bar{n}} = 0$  the condition  $\boxed{\Psi_u = -\bar{n}(p - \bar{p}_i)}$  is obtained as boundary condition for the shape to be modified.

The values of  $\Psi_u$  are negligible as long as we are far from the hole inlet, so without violation, can be considered equal to zero at the inlet and exit. At the rest of the walls, the part which is not being modified also can be considered zero as an adjoint “no-slip” condition. For  $\Psi_p$  a Neumann condition is applied, namely,  $\nabla \Psi_p \cdot \bar{\mathbf{n}} = 0$  for the entire boundary except from the symmetric boundaries where for both  $\Psi_u$  and  $\Psi_p$  symmetry conditions are applied.

### 3.9.2 Given $C_d$ value

In order to explore the potential of the method, a suitable cost function for optimised injector design has been introduced. The discharge coefficient ( $C_d$ ) has been used as the means to control cavitation inception within the nozzle hole. This case is also challenging with respect to the adjoint analysis. The complications in the application of this design approach arise from the fact that the cost function is defined on the hole exit whilst

the gradient and the geometry under modification is actually the hole inlet. To overcome these complications appropriate boundary conditions must be enforced in the area of the wall boundary under modification.

Considering and applying variation to the cost function (3.5) defined in section 3.4 equation (3.49) takes the following form for the hole exit:

$$\begin{aligned} & \frac{(\bar{C}_d - C_d)}{A_{exit} \cdot \sqrt{\frac{2(p_{in} - p_{out})}{\rho}}} \oint_{exit} \delta \mathbf{u} \bar{\mathbf{n}} ds = \\ & = \oint_{exit} \Psi_p \cdot \rho \delta \mathbf{u} \bar{\mathbf{n}} ds + \oint_{exit} \Psi_u \cdot (\rho \delta \mathbf{u} \otimes \mathbf{u}) \bar{\mathbf{n}} ds + \oint_{exit} \Psi_u \cdot (\rho \mathbf{u} \otimes \delta \mathbf{u}) \bar{\mathbf{n}} ds \end{aligned} \quad (3.54)$$

where  $\bar{C}_d$  is the desired value of the  $C_d$  that we want to achieve,  $\rho$  is the density,  $\mathbf{u}$  is the velocity vector,  $\bar{\mathbf{n}}$  is the normal unit vector along the boundary as before  $p_{in}$  and  $p_{out}$  are the pressures at the inlet and outlet respectively and  $A_{exit}$  is the area at the exit. The  $C_d$  takes values from 0 to 1 the last being the less cavitating case. Equation (3.54) can be written in differential form as:

$$\Psi_u \cdot [(\rho \delta \mathbf{u} \otimes \mathbf{u}) + (\rho \mathbf{u} \otimes \delta \mathbf{u})] + \Psi_p \delta \mathbf{u} = - \frac{(\bar{C}_d - C_d)}{A_{exit} \cdot \sqrt{\frac{2(p_{in} - p_{out})}{\rho}}} \delta \mathbf{u} \quad (3.55)$$

Where for arbitrary values of  $\delta \mathbf{u}$  and Cartesian reference frame the expression in brackets can be calculated to give:

$$\begin{aligned} & (\rho \delta \mathbf{u} \otimes \mathbf{u}) + (\rho \mathbf{u} \otimes \delta \mathbf{u}) = \\ & \begin{pmatrix} \delta(\rho u)u & \delta(\rho v)u & \delta(\rho w)u \\ \delta(\rho u)v & \delta(\rho v)v & \delta(\rho w)v \\ \delta(\rho u)w & \delta(\rho v)w & \delta(\rho w)w \end{pmatrix} + \begin{pmatrix} (\rho u)\delta u & (\rho v)\delta u & (\rho w)\delta u \\ (\rho u)\delta v & (\rho v)\delta v & (\rho w)\delta v \\ (\rho u)\delta w & (\rho v)\delta w & (\rho w)\delta w \end{pmatrix} = \\ & = \rho \begin{pmatrix} 2u\delta u & v\delta u + u\delta v & w\delta u + u\delta w \\ u\delta v + v\delta u & 2v\delta v & w\delta v + v\delta w \\ u\delta w + w\delta u & v\delta w + w\delta v & 2w\delta w \end{pmatrix} = (\rho \mathbf{u} + \rho \mathbf{u}^T) \delta \mathbf{u} \end{aligned} \quad (3.56)$$



Replacing the result (3.56) in equation (3.55) an expression independent from  $\delta \mathbf{u}$  is obtained:

$$\Psi_{\mathbf{u}} \cdot (\rho \mathbf{u} + \rho \mathbf{u}^T) + \Psi_p = - \frac{(\bar{C}_d - C_d)}{A_{exit} \cdot \sqrt{\frac{2(p_{in} - p_{out})}{\rho}}} \quad (3.57)$$

Inversing  $\mathbf{B} = \rho \mathbf{u} + \rho \mathbf{u}^T$  we obtain:

$$\Psi_{\mathbf{u}} = - \left( \frac{(\bar{C}_d - C_d)}{A_{exit} \cdot \sqrt{\frac{2(p_{in} - p_{out})}{\rho}}} + \Psi_p \right) \mathbf{B}^{-1} \quad (3.58)$$

Also in this case we can assume that the values of  $\Psi_{\mathbf{u}}$  are negligible as far as we are from the sharp hole inlet so without violation we can set them equal to zero at the inlet of the nozzle. For  $\Psi_p$  a Neumann condition is applied  $\nabla \Psi_p \cdot \vec{n} = 0$  for the inlet and outlet. In the symmetric boundaries again for both  $\Psi_{\mathbf{u}}$  and  $\Psi_p$  symmetry conditions are applied.

In this optimisation case where the cost function is defined on the exit boundary the value of its gradients on the wall are eliminated. By substituting this zero gradients in equation (3.49) result in Dirichlet boundary conditions for  $\Psi_{\mathbf{u}}$  on the wall equivalent to  $\Psi_{\mathbf{u}} \cdot \vec{n} = 0 \Leftrightarrow \Psi_{\mathbf{u}} = 0$ . Choi [183] suggested this boundary condition in the similar case of pressure drop minimisation. The author believes that this is not possible in this case. The wall in this case is the moving boundary so  $\Psi_{\mathbf{u}} = 0$  is replaced directly in the gradients of the cost function with respect to the design variables for the third condition (3.22). The most important terms in those gradients are the terms involving  $\Psi_{\mathbf{u}}$  [146]. For this reason Neumann condition is applied  $\nabla \otimes \Psi_{\mathbf{u}} = 0$  and Dirichlet for  $\Psi_p = 0$ .

### 3.10 FORMULATION OF GRADIENTS

In the previous paragraphs the adjoint equations have been derived using the second condition equation (3.21). Having the adjoint equations being solved and the costate variables obtained, the sensitivity derivatives can be easily computed. The values of the costate variables can be substituted to the third condition equation (3.22) for  $I_{aug}$  giving the adjoint gradients with respect to the design variables  $D$ . The gradients or else sensitivity derivatives consist of the boundary distribution of the costate variables and their derivatives combined with terms arising from  $\delta I_C$  plus the remaining terms resulting from the diffusive part of the adjoint system calculation that multiplies the variation of the velocities  $\delta u$ ,  $\delta v$  and  $\delta w$ . So the variation of  $I_{aug}$  reads:

$$\delta I_{aug} = \frac{\partial I_C}{\partial D} \delta D + \oint_W p \Psi_u \delta(\bar{n} ds) + \oint_W \bar{n} \mu \left[ \nabla \otimes \Psi_u + (\nabla \otimes \Psi_u)^T \right] \delta \mathbf{u} \quad (3.59)$$

The rest of the boundary terms are equal to zero due to the no-slip boundary condition enforced on the wall. The geometry under modification in this study is part of the wall boundary in both optimisation cases.

Following the analysis proposed by Anderson [7] the term of equation (3.59), resulting from the diffusion part of the adjoint equations, can be expressed in terms of design variables. Expressing the velocities on the new surface in a Taylor series and noting that the velocities on the old and new surface are both zero, the variation of the velocity components can be written in the following manner that contribute to the sensitivity derivatives:

$$\delta \mathbf{u} = -\frac{\partial \mathbf{u}}{\partial x} \delta x - \frac{\partial \mathbf{u}}{\partial y} \delta y - \frac{\partial \mathbf{u}}{\partial z} \delta z \quad (3.60)$$

The variational expressions of  $I_{aug}$  can be also significantly simplified by the substitution of the normal vector's definition:

$$\delta(\bar{n} ds) = \delta((n_x, n_y, n_z) ds) = \delta\left(\left(\frac{dx}{ds}, \frac{dy}{ds}, \frac{dz}{ds}\right) ds\right) = (\delta(dx), \delta(dy), \delta(dz)) \quad (3.61)$$

Considering the design points to be  $D_i$  where  $i = 1, \dots, n$ , the variations of the geometrical quantities in  $\delta I_{aug}$  read:

$$\begin{aligned}
\delta(ds) &= \sum_{i=1}^n \frac{\partial(ds)}{\partial D_i} \delta D_i, \\
\delta(dx) &= \sum_{i=1}^n \frac{\partial(dx)}{\partial D_i} \delta D_i, \\
\delta(dy) &= \sum_{i=1}^n \frac{\partial(dy)}{\partial D_i} \delta D_i, \\
\delta(dz) &= \sum_{i=1}^n \frac{\partial(dz)}{\partial D_i} \delta D_i.
\end{aligned} \tag{3.62}$$

After replacing the variations  $\delta(dx)$ ,  $\delta(dy)$ ,  $\delta(dz)$  and  $\delta(ds)$  of  $\delta I_{aug}$ , the expression for the sensitivity derivatives is obtained. The final form of the sensitivity derivatives with respect to the design variables is obtained using the shape parameterisation and the consequence derivation of the variations of the above geometrical entities. The variations of the geometric entities using a central finite difference scheme around a small perturbation  $\epsilon$ , for example for  $\delta(ds)$  we have:

$$\delta(ds) = \frac{ds(\delta D_{\epsilon}) - ds(\delta D_{-\epsilon})}{\epsilon} \tag{3.63}$$

For the rest of the geometric entities the same formula is employed.

The only term that needs to be calculated is the gradient of the cost function  $I_C$  on the wall. For the inverse case of the prescribed target pressure, the gradient expression is:

$$\frac{\partial I_C}{\partial D} \delta D = \frac{1}{2} \oint_w (p - \bar{p}_t)^2 \delta(ds) \tag{3.64}$$

While for the  $C_d$  case, the gradient expression is:

$$\frac{\partial I_C}{\partial D} \delta D = 0 \tag{3.65}$$

So the design parameters  $D$  are updated using the optimisation method described in the next section and the whole process is repeated.

### 3.11 UPDATING THE DESIGN VARIABLES

The last thirty years has seen the development of a powerful collection of algorithms for the unconstrained optimisation of smooth functions. All algorithms for unconstrained minimisation require the user to supply a starting point, which in our case, is the set of parameters that represents the initial geometry. Knowledge of the physics about the problem can help the user to choose a starting geometry that will be reasonably close to the solution. Otherwise the starting geometry must be chosen in an arbitrary manner.

Optimisation algorithms begin from the initial geometry and generate a sequence of intermediate results, in an iterative process that terminates when either no more progress can be made or when it seems that a solution has been approximated with sufficient accuracy. In deciding how to move from the one iteration to the next, the algorithm uses information about the cost function  $I_c$  at the current iteration. The method used in this study is a simple line search method called steepest descent which uses the information of the first gradient of the cost function.

The direction of the negative gradient is indeed the most obvious search direction for line search method because it is the direction where the cost function decreases most rapidly. This is easily proven by use of the Taylor theorem [35]. There are line search methods using both the first and the second derivative of the cost function but since the adjoint method cannot provide these in straightforward manner, such a method is not convenient in our case. The steepest descent method doesn't need this kind of information but can be rather slow in some cases. Of course there are other methods which use only information of the first gradient of the cost function which are not explored in this study. The reason is that the focus of this study is the extraction of the gradients using the adjoint method and not a research on efficiency of the optimisation method itself; moreover the optimisation method can be easily replaced in the code and also existing libraries with optimisation methods can be used.

The steepest descent method moves along the negative gradient direction at every step. After the final form of the sensitivity derivatives with respect to the design variables is concluded, the update of the control points location is carried out through the use of steepest descend scheme, reading:

$$D_i^{new} = D_i - \eta \frac{\partial I_{aug}}{\partial D_i} \quad (3.66)$$

The previous expression stands for positive  $\eta$  values ( $\eta$  being the step of the descent method). By using the gradient of the cost function, the search on the design space is limited to a specific direction, rather than randomly searching the entire  $n$ -dimensional space ( $n$  being the number of parameters).

There are various ways to choose the value of the step size  $\eta$ . In this study the choice was to first non-dimensionalise the  $\eta$  values by dividing with the normalised gradient; this reads:

$$\left\| \frac{\partial I_{aug}}{\partial D_i} \right\| = \sqrt{\sum_{i=1}^n \left( \frac{\partial I_{aug}}{\partial D_i} \right)^2} \quad (3.67)$$

After that, a good estimate for  $\eta$  is usually an order of magnitude less than the value of each design variable  $D_i$ . A proper estimation of the scalar parameter  $\eta$  provides the minimum for the current search direction. In this study due to the nature of the parameterisation method used the value of  $\eta$  depends on the parameter and is defined as  $\eta(D)$ . This is necessary due to the difference in the magnitude of the different parameters describing the nozzle shape. Vrahatis et al [202] has used individual  $\eta$  values and observed significant improvement in the convergence. Moreover they also used an adaptive time step which is not implemented in the present study. More about descent methods can be found in [55].

### 3.12 SUMMARY

The setup of the problem that we are dealing with has been discussed in detail throughout this chapter. The iterative nature of the optimisation method was described and every step of this iterative process was discussed in detail. The parameterisation method developed for this study was analysed as well as the grid modification method. The adjoint equations of the Navier Stokes flow solver have been extracted along with the corresponding boundary conditions. Finally the gradients with respect to the parameters are calculated and driven to zero using the steepest descent method. In the next chapter the discretisation and solution procedure of the adjoint equations are cross referenced with the ones of the incompressible Navier-Stokes flow equations.

## *Chapter 4*

# *DISCRETISATION AND SOLUTION OF THE FLOW AND ADJOINT EQUATIONS*

### **4.1 INTRODUCTION**

The control volume approach has been adopted in this study, for the analysis of the fluid flow properties and subsequently for the adjoint field properties as well. The solution procedure for the flow in the continuous phase in conjunction with the adjoint equations are presented in this chapter. The in-house **GFS** software, which employs an unstructured-grid RANS flow solver, has been used to calculate the flow on this study. The adjoint solver has been added to this code and uses the same methods as the flow solver. The discretisation of the generic equation will be used as the origin for the discretisation and solution of both the flow and adjoint equations. The finite volume discretisation method will be described; this method is applied for the numerical solution of the governing continuous phase equations and subsequently for the adjoint equations.

### **4.2 THE GFS CODE FOR COMPUTING THE FLOW FIELD**

The **GFS** code has been used for the flow calculations during this study. The aforementioned code is an unstructured-grid RANS finite volume incompressible flow solver and has been used for the calculation of the single-phase internal nozzle flow distribution, using a pressure correction methodology. The method uses a finite-volume formulation of the Navier-Stokes equations which takes the conservation laws written in integral form discretised about control volumes. One advantage of this method is that the fluxes can be converted to surface integrals via Gauss-divergence theorem. Finite-volume schemes are much more suited for solving complicated geometrical problems due to the fact that they can be solved directly in the physical domain. Discretisation in the physical space ensures that mass and momentum are conserved also on the discrete level [203]. Various differentiation schemes and solvers are used for the solution of the mo-

momentum equations and the corresponding adjoint transport equation. All these methods are explained in more details in the next sections.

### 4.3 THE GENERIC TRANSPORT EQUATION

The fluid flow is mathematically described by three conservation laws, namely, the conservation of mass, momentum and energy. These laws completely determine the physical behaviour of the fluid and are totally independent of the nature of the fluid, which is defined by additional properties such as viscosity, heat conductivity, surface tension and compressibility.

Let us consider the integral form of the generic transport equation for any scalar or tensorial quantity  $\phi$  [204], which is valid for any control volume  $V$  with boundary  $\partial V$ :

$$\underbrace{\frac{\partial}{\partial t} \int_V \rho \phi dV}_{\text{time derivative}} + \underbrace{\oint_{\partial V} \rho \phi \mathbf{u} \cdot d\mathbf{S}}_{\text{convection term}} = \underbrace{\oint_{\partial V} \Gamma \nabla \phi d\mathbf{S}}_{\text{diffusion term}} + \underbrace{\int_V S_\phi(\phi) dV}_{\text{source term}} \quad (4.1)$$

where  $\rho$  is the density,  $\mathbf{u}$  is the velocity vector,  $\Gamma$  is the diffusivity,  $S_\phi$  represents a source term and  $d\mathbf{S}$  is the outward pointing face vector. The various terms in the above equation represent the rate of change per unit volume, the efflux by convection per unit volume, the rate of diffusion and the rate of production/destruction per unit volume.

By virtue of Gauss's divergence theorem a surface integral can be converted into volume integral. Its application for example to equation (4.1) leads to the following equivalent form of the generic transport equation:

$$\frac{\partial}{\partial t} \int_V \rho \phi dV + \int_V \nabla \cdot (\rho \phi \mathbf{u}) dV = \int_V \nabla \cdot (\Gamma \nabla \phi) dV + \int_V S_\phi(\phi) dV \quad (4.2)$$

For an infinitesimal volume whose size tends towards zero the conservative differential form of the generic transport equation can be derived from the above:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi(\phi) \quad (4.3)$$

Due to fact that there is not only spatial but also temporal variation of  $\phi$ , the generic transport equation needs to be integrated in time as well. With integration of equation (4.3) in time and over the control volume of the following equation is derived:

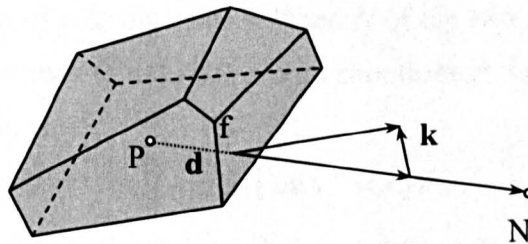
$$\begin{aligned}
& \int_t^{t+\Delta t} \left[ \int_{V_P} \frac{\partial \rho \varphi}{\partial t} dV + \int_{V_P} \nabla \cdot (\rho \varphi \mathbf{u}) dV \right] dt \\
& = \int_t^{t+\Delta t} \left[ \int_{V_P} \nabla \cdot (\Gamma \nabla \varphi) dV + \int_{V_P} S_\varphi(\varphi) dV \right] dt
\end{aligned} \tag{4.4}$$

where  $\Delta t$  is the numerical time step, whose value affects the accuracy of the numerical time integration. The solution of  $\varphi$  at time level  $t$  is assumed to be the initial condition thus a known initial condition. Equation (4.4) will serve as the basis for the description of the discretisation approach. The discretisation of the Navier-Stokes and adjoint equations results naturally from the analysis.

#### 4.4 CALCULATION OF BASIC ALGEBRAIC EXPRESSIONS

The equations of fluid mechanics are solvable for only a limited number of flows. The analytic solutions are extremely useful in helping to understand fluid flow but rarely can they be used directly in engineering analysis or design. That is the case for the Navier-Stokes equations which cannot be solved analytically so an approximate solution through the use of a numerical method is pursued. For fluid flow equations the most widely used is the so-called finite volume method [205]. In order to apply this method, the solution domain is converted into a computational mesh that comprises small spatial subdivisions of the domain, the cells or control volumes (CV). The solution domain is or encloses the geometry of interest, depending on whether an internal or external flow is to be simulated.

An implementation of the finite volume method which uses a face addressing technique is used as a basis for this study. The objective of the face addressing technique is to allow the use of arbitrary shaped cells whilst preserving the efficiency of the computational algorithm.



**Figure 4.1: Example of control volume**



The above figure represents a typical control volume (cell) with an arbitrary number of faces. The point  $P$  is the centre of the control volume of interest. By  $N$  the centre of the adjacent control volume has been designated. The vector  $\mathbf{d} = \overrightarrow{PN}$  connects these two centres and  $\mathbf{S}$  is the outward-pointing face area vector that is vertical to the face. The mesh is termed non-orthogonal when the angle between  $\mathbf{d}$  and  $\mathbf{S}$  is not zero. In order to account for the orthogonal and non-orthogonal contributions to the face gradients, the vectors  $\mathbf{k}$  and  $\mathbf{D}$  are introduced. Following Jasak [206], the over relaxed approach is chosen for splitting the non-orthogonal contribution  $\mathbf{k}$  from the orthogonal  $\mathbf{D}$ ; this approach was demonstrated to be superior in various numerical tests [206]. This approach will be used here in the discretisation of the conservation equations. The two aforementioned vectors are defined according to the following equations:

$$\mathbf{D} = \frac{\mathbf{d}}{\mathbf{d} \cdot \mathbf{S}} |\mathbf{S}|^2$$

$$\mathbf{k} = \mathbf{S} - \mathbf{D} \quad (4.5)$$

After the discretisation of space the application of the finite volume method is possible, with which the discretised form of the equations that are to be solved will be constructed.

#### 4.4.1 Face Interpolation

One of the most fundamental issues within the context of the finite volume method is the numerical interpolation that is employed for the calculation of the face variable values from the corresponding cell-centred ones. The selection of a non-suitable interpolation scheme can have disastrous consequences on the numerical accuracy and in some cases even on the stability of the method. On the contrary, a robust and accurate scheme can ensure stability and moreover increase the efficiency of the algorithm.

Let us consider a scalar  $\phi$  whose value at the face  $f$  of the control volume needs to be estimated, from values of  $\phi$  at the centres  $P$  and  $N$  of the two adjacent control volumes. Before proceeding, we need to define the mass flux through face  $f$ , as it is necessary for some interpolation schemes:

$$C_f = \mathbf{S}_f \cdot (\rho \mathbf{u})_f = \rho_f F_f \quad (4.6)$$

where  $F_f$  is the volume flux. For the time being and since the discussion is about how to calculate the face value of  $\phi$ , the face values of the velocity are assumed to be known. The face interpolation schemes that have been used in this study are the following:

### Central differencing (CD)

The first and most straightforward thing one would attempt is to use linear interpolation for calculating the face value:

$$\varphi_{f(CD)} = f_P \varphi_P + (1 - f_P) \varphi_N \quad (4.7)$$

$$\text{where } f_P = \frac{|\vec{fN}|}{|\vec{fN}| + |\vec{Pf}|}$$

This scheme is second-order accurate, which is considered satisfactory, but when it is used to interpolate face values for the discretised convection term of equation (4.4), under certain circumstances, it can lead to unbounded solutions and numerical divergence [204]. When the boundedness of the interpolated scalar/vector is guaranteed, for example if  $\varphi$  was the density or a phase fraction, it can be used without problems. Moreover, it is used for interpolating discretised differential operators, *e.g.* the gradient or the divergence of a variable.

### Upwind differencing (UD)

This scheme is directly related to the convective character of fluid flows, and for this reason the direction of the flow is essential:

$$\varphi_{f(UD)} = \begin{cases} \varphi_P & \text{for } C_f \geq 0 \\ \varphi_N & \text{for } C_f < 0 \end{cases} \quad (4.8)$$

UD is a first order accurate scheme, which can affect adversely the accuracy of the solution; nevertheless it guarantees its boundedness, a feature that makes this scheme a valuable component of any CFD code based on the finite volume method.

### Blended differencing (BD)

There are schemes that combine UD and CD in an effort to create a scheme with a reasonable trade-off between accuracy and boundedness. The blending differencing (BD) scheme is created in this way and is the scheme used on this study. This combination of UD and CD is linear, according to the following equation:

$$\varphi_{f(BD)} = \gamma \varphi_{f(CD)} + (1 - \gamma) \varphi_{f(UD)} \quad (4.9)$$

The blending factor  $0 \leq \gamma \leq 1$  determines how much numerical diffusion will be introduced. Several theoretical attempts have been made in order to calculate this blending factor in the most efficient way [204]. In this study the Gamma differencing scheme that

was introduced by Jasak et al [207] has been utilised. This scheme has the advantage that has been developed for unstructured meshes and it is second-order accurate.

#### 4.4.2 Differential Operators

Before proceeding to the discretisation of the various terms of the generic equation some additional operators need to be defined as presented in [208]. In the discretised form of the equations some differential operators, such as the gradient of a scalar, appear quite often; for this reason their discretisation will be presented separately in this paragraph. By applying the theorem of Gauss, these operators can be discretised in the following fashion:

##### Scalar gradient

$$\int_V \nabla \phi dV = \oint_{\partial V} d\mathbf{S} \phi = \sum_{f=1}^m \int_f d\mathbf{S} \phi \simeq \sum_{f=1}^m \mathbf{S}_f \phi_f \quad (4.10)$$

##### Vector/Tensor divergence

$$\int_V \nabla \cdot \boldsymbol{\phi} dV = \oint_{\partial V} d\mathbf{S} \cdot \boldsymbol{\phi} = \sum_{f=1}^m \int_f d\mathbf{S} \cdot \boldsymbol{\phi} \simeq \sum_{f=1}^m \mathbf{S}_f \cdot \boldsymbol{\phi}_f \quad (4.11)$$

##### Vector/Tensor gradient

$$\int_V \nabla \otimes \boldsymbol{\phi} dV = \oint_{\partial V} d\mathbf{S} \otimes \boldsymbol{\phi} = \sum_{f=1}^m \int_f d\mathbf{S} \otimes \boldsymbol{\phi} \simeq \sum_{f=1}^m \mathbf{S}_f \otimes \boldsymbol{\phi}_f \quad (4.12)$$

where  $m$  is the total number of faces of the control volume and  $\phi_f$  is the interpolated value of  $\phi$  at face  $f$ . Although the latter can be evaluated with any of the aforementioned schemes described in the previous section, for the discretised equations of the above operators CD interpolation is used exclusively throughout this study. The above expressions of the differential operators will be employed in the forthcoming discretisation of the convection and diffusion terms of the generic transport equation. Equations (4.10) to (4.12) can also be used to calculate a second-order accurate approximation of the gradient of a flow variable at the cell centre of a control volume:

$$(\nabla \varphi)_P \simeq \frac{1}{V_P} \sum_{f=1}^m \mathbf{S}_f \varphi_f \quad (4.13)$$

$$(\nabla \cdot \boldsymbol{\varphi})_P \simeq \frac{1}{V_P} \sum_{f=1}^m \mathbf{S}_f \cdot \boldsymbol{\varphi}_f \quad (4.14)$$

$$(\nabla \otimes \boldsymbol{\varphi})_P \simeq \frac{1}{V_P} \sum_{f=1}^m \mathbf{S}_f \otimes \boldsymbol{\varphi}_f \quad (4.15)$$

where  $P$  is the centre of the control volume depicted in and  $V_P$  is its actual volume. For the evaluation of the above gradients at faces CD interpolation is used again. Nevertheless, if we move to the faces all vector and tensor variables can be decomposed into orthogonal and non-orthogonal contributions with the over-relaxed treatment. For gradients in particular though, their orthogonal component can be evaluated directly. If the control volume of Figure 4.1 is considered, then the following equation is valid for the orthogonal component of the gradient  $(\nabla^\perp)$  of scalar  $\varphi$ :

$$\nabla_f^\perp \varphi = \frac{\varphi_N - \varphi_P}{|\mathbf{d}|} \quad (4.16)$$

The complete face gradient can be decomposed into its orthogonal and non-orthogonal contributions with the following equation:

$$\frac{\mathbf{S}_f \cdot \nabla_f \varphi}{|\mathbf{S}_f|} = \underbrace{\frac{|\mathbf{D}_f|}{|\mathbf{S}_f|} \frac{\varphi_N - \varphi_P}{|\mathbf{d}|}}_{\text{orthogonal}} + \underbrace{\frac{\mathbf{k}_f \cdot (\nabla \varphi)_f}{|\mathbf{S}_f|}}_{\text{non-orthogonal}} \quad (4.17)$$

where  $(\nabla \varphi)_f$  is evaluated with CD interpolation from  $(\nabla \varphi)_P$  and  $(\nabla \varphi)_N$ . This decomposition is utilised in the forthcoming discretisation procedure for numerical reasons.

#### 4.4.3 Time derivative and Temporal Approximation

Although in this study the problems that are solved are all steady state, the temporal discretisation is implemented in the developed software and it is presented here for completeness. The discretisation of the time derivate of equation (4.4) is straightforward after the integration over the control volume is performed. When a time dependant prob-

lem is simulated, the term  $\int_{dV} \frac{\partial \rho \varphi}{\partial t} dV$  is also present. When a first order accurate

discretisation is used, the above term can be approximated by assuming linear variation of  $\varphi$  within a time step as:

$$\int_{V_P} \frac{\partial \rho \varphi}{\partial t} dV = \frac{(\rho \varphi V)_P^{t+\Delta t} - (\rho \varphi V)_P^t}{\Delta t} \quad (4.18)$$

One issue that needs discussion at this point is the time level approximation. All terms of equation (4.4) are time-dependent, which means that they have to be approximated at some point between time levels  $t$  and  $t+\Delta t$ . Let us consider a representation of the differential form of the generic transport equation in the following way:

$$\frac{\partial \rho \varphi}{\partial t} = f(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi) \quad (4.19)$$

If equation (4.19) is integrated in time and over the control volume we obtain:

$$\int_t^{t+\Delta t} \left[ \int_{V_P} \frac{\partial \rho \varphi}{\partial t} dV \right] dt = \int_t^{t+\Delta t} \left[ \int_{V_P} f(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi) dV \right] dt \quad (4.20)$$

which becomes according to equation (4.18):

$$\frac{(\rho \varphi V)_P^{t+\Delta t} - (\rho \varphi V)_P^t}{\Delta t} = \int_t^{t+\Delta t} g(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi) dt \quad (4.21)$$

where  $g(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi) = \int_{V_P} f(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi) dV$

The r.h.s. of the above equation has to be approximated with respect to time; by introducing an assumption about the variation of function  $g$  over the time step  $\Delta t$ , the following expression can be derived:

$$\begin{aligned} & \frac{(\rho \varphi V)_P^{t+\Delta t} - (\rho \varphi V)_P^t}{\Delta t} \\ &= (1 - \beta) [g(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi)]^t + \beta [g(\rho, \varphi, \mathbf{u}, \Gamma, S_\varphi)]^{t+\Delta t} \end{aligned} \quad (4.22)$$

where  $\beta$  is a weighting factor that effectively controls the type of time integration scheme that is employed.

Among numerical algorithms the most popular choices for  $\beta$  are  $\beta = 0$  for the explicit and  $\beta = 1$  for the implicit Euler schemes, or  $\beta = 0.5$  for the implicit Crank-Nicolson scheme. The Euler schemes are first order accurate whereas the Crank-Nicolson is second order. The latter is an implicit scheme too; both the implicit Euler and the Crank-

Nicolson schemes are unconditionally stable, whereas for the explicit Euler stability restrictions apply.

#### 4.4.4 Convection Term

We come now to the discretisation of the convection term of equation (4.1). As already discussed, by employing Gauss's theorem the integral over the control volume can be transformed into a surface integral as follows:

$$\int_{V_P} \nabla \cdot (\rho \phi \mathbf{u}) dV = \oint_{\partial V_P} \rho \phi \mathbf{u} \cdot d\mathbf{S} \simeq \sum_{f=1}^m \mathbf{S}_f \cdot (\rho \mathbf{u})_f \phi_f = \sum_{f=1}^m \rho_f F_f \phi_f \quad (4.23)$$

where  $F_f$  is the volume flux through face  $f$  and is defined as  $F_f = \mathbf{S}_f \cdot \mathbf{u}_f$ . Again, the value  $\phi_f$  on face  $f$  can be evaluated with the methods described in Section 4.4.1. Albeit, there is an important detail that needs to be mentioned; for the evaluation of the face velocities a special interpolation technique has to be utilised, which will be described together with the derivation of the pressure equation in Section 4.5.3.

#### 4.4.5 Diffusion Term

The discretisation of the diffusion term of equation (4.1) is done in the same fashion; by employing Gauss's theorem the volume integral is converted into a surface one:

$$\int_{V_P} \nabla \cdot (\Gamma \nabla \phi) dV = \oint_{\partial V_P} d\mathbf{S} \cdot \Gamma \nabla \phi \simeq \sum_{f=1}^m \Gamma_f \mathbf{S}_f \cdot \nabla_f \phi \quad (4.24)$$

The above approximation is only valid if  $\Gamma$  is a scalar variable. A detail regarding equation (4.24) is that evaluation of  $\nabla \phi$  at face  $f$  is needed. In order to evaluate this,  $\nabla \phi$  is calculated at the centres  $P$  and  $N$  of the two adjacent control volumes in Figure 4.1: Example of control volume by using equation (4.13), and then CD interpolation is utilised to obtain the value of  $\nabla_f \phi$ .

#### 4.4.6 Source Term

The last part of the general transport equation that needs discretisation is the source term. Generally speaking, the source term, defined as  $S_\phi(\phi)$ , can be any arbitrary function of  $\phi$ . Nevertheless, it is necessary to be linearised before discretisation:

$$S_\phi(\phi) = \phi S_P + S_U \quad (4.25)$$

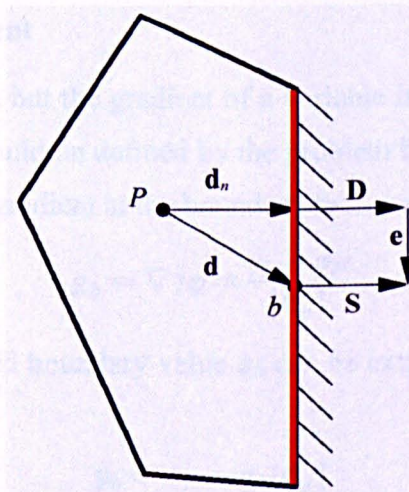
where  $S_P$  and  $S_U$  may depend on  $\phi$ . Integration of equation (4.25) over the control volume of interest yields:

$$\int_{V_P} S_\phi(\phi) dV \simeq \int_{V_P} (\phi S_P + S_U) dV \simeq \phi_P S_P V_P + S_U V_P \quad (4.26)$$

The linearisation of source terms is quite important, because it gives the possibility to split the contribution of the original source term to the final discretised system of algebraic equations into an explicit and an implicit component. This latter numerical detail can enhance or in some cases even deteriorate, depending on correct or wrong choice of linearisation, the numerical stability and boundedness of the solution algorithm.

#### 4.4.7 Boundary Conditions

Before arriving to the final form of the discretised equations, one last issue has to be addressed; the implementation of the physical boundary conditions. As physical boundary conditions we refer to the actual boundary conditions of the mathematical problem prior to discretisation. The face-based approach, which has been used for the discretisation of the various equation terms, has depended so far on the fact that each face inside the solution domain is always found between two adjacent control volumes. Nevertheless, the faces of the control volumes that are located at the solution domain boundaries are one-sided. For this reason ‘pseudo-cells’ are introduced at the boundaries. The ‘pseudo-cells’ are the cells that have no volume and effectively have the same geometrical characteristics like the boundary faces. With the pseudo-cells the physical boundary conditions of the problem are realised numerically.



**Figure 4.2: Control volume with a boundary face. The red line depicts the “pseudo-cell”**

An example of such a control volume is shown in Figure 4.2, in which some additional parameters have been provided. Again the vector between cell centre  $P$  and pseudo-cell centre  $b$ , which coincides with the centre of the boundary face, is denoted by  $\mathbf{d}$ , and its component normal to the face by  $\mathbf{d}_n$ . The latter is calculated by the following equation:

$$\mathbf{d}_n = \frac{\mathbf{d} \cdot \mathbf{S}}{|\mathbf{S}|^2} \mathbf{S} \quad (4.27)$$

Since it is assumed that the boundary condition to be applied is valid along the whole of the face, and that vector  $\mathbf{e}$  is relatively small, we have that vector  $\mathbf{D} = \mathbf{S}$ . Generally speaking, there two-types of numerical boundary conditions that are applied:

### Dirichlet or Fixed Value

With this boundary condition the value of the dependent variable is prescribed or ‘fixed’ at the boundary face, *i.e.*  $\varphi_f = \varphi_b$ . Nevertheless, because of some terms also the gradient  $\nabla_f \varphi$  needs to be evaluated at the boundary faces. In that case we take advantage of the fact that the boundary condition is valid for the whole of the face, which allows for the calculation of only the orthogonal contribution of the gradient. For example, as already presented in Section 4.4.1, calculation of  $\mathbf{S}_f \cdot \nabla_f \varphi$  is necessary for the discretisation of the diffusion term at the boundary. From equations (4.17) and (4.24) we have:

$$\Gamma_f \mathbf{S}_f \cdot \nabla_f \varphi = \Gamma_f |\mathbf{S}_f| \frac{\varphi_b - \varphi_P}{|\mathbf{d}_n|} \quad (4.28)$$

In equation (4.28) the fact that  $\mathbf{D} = \mathbf{S}$  has been taken into account.

### Neumann or Fixed Gradient

In some cases not the value but the gradient of a variable is prescribed at the boundary; for example the heat flux could be defined by the problem but not the temperature itself. Supposedly  $g_b$  is the given gradient at the boundary face, we then have:

$$g_b = \nabla_f^\perp \varphi = \frac{\varphi_b - \varphi_P}{|\mathbf{d}_n|} \quad (4.29)$$

which means that the needed boundary value  $\varphi_b$  can be extrapolated to the boundary by using the gradient  $g_b$ :

$$\varphi_b = \varphi_P + g_b |\mathbf{d}_n| \quad (4.30)$$



#### 4.4.8 System of Linear Algebraic Equations

In this section all the previously discretised parts of the generic transport equation will be assembled, in order to form the final set of algebraic equations that need to be solved. The initial form of the transport equation from which the discretisation started was equation (4.4). If equations (4.18), (4.23), (4.24) and (4.26) are combined with the above equation, then the following expression is derived:

$$\begin{aligned} & \int_t^{t+\Delta t} \left[ \frac{(\rho\phi V)_P^{t+\Delta t} - (\rho\phi V)_P^t}{\Delta t} + \sum_{f=1}^m \rho_f F_f \phi_f \right] dt \\ &= \int_t^{t+\Delta t} \left[ \sum_{f=1}^m \Gamma_f \mathbf{S}_f \cdot \nabla_f \phi + \phi_P S_P V_P + S_U V_P \right] dt \end{aligned} \quad (4.31)$$

This form of the transport equation is referred to as *semi-discretised* [209]. Recalling equation (4.22), the following equation is formulated:

$$\begin{aligned} & \frac{(\rho\phi V)_P^{t+\Delta t} - (\rho\phi V)_P^t}{\Delta t} + (1-\beta) \left[ \sum_{f=1}^m \rho_f F_f \phi_f \right]^t + \beta \left[ \sum_{f=1}^m \rho_f F_f \phi_f \right]^{t+\Delta t} \\ &= (1-\beta) \left[ \sum_{f=1}^m \Gamma_f \mathbf{S}_f \cdot \nabla_f \phi + \phi_P S_P V_P + S_U V_P \right]^t \\ &+ \beta \left[ \sum_{f=1}^m \Gamma_f \mathbf{S}_f \cdot \nabla_f \phi + \phi_P S_P V_P + S_U V_P \right]^{t+\Delta t} \end{aligned} \quad (4.32)$$

As already mentioned,  $\beta$  is a weighting factor with value range  $0 \leq \beta \leq 1$ , which controls the time integration scheme that is employed. The Euler and Crank-Nicolson schemes that are implemented in the GFS code correspond to values of  $\beta = 1$  and 0.5 respectively. For simpler notation the new time level  $\phi^{t+\Delta t}$  is denoted as  $\phi^n$  and the old time level  $\phi^t$  as  $\phi^o$ . Equation (4.32) with aid of equation (4.17) can be re-arranged into the following form, which is one step before the linear system formulation:

$$\begin{aligned} & \frac{\rho_P^n V_P^n}{\Delta t} \phi_P^n - \beta S_{P_P}^n V_P^n \phi_P^n + \beta \sum_{f=1}^m \rho_f^n F_f^n \phi_f^n - \beta \underbrace{\sum_{f=1}^m \Gamma_f^n \mathbf{D}_f \cdot (\nabla \phi)_f^n}_{\text{orthogonal}} \\ &= \frac{\rho_P^o V_P^o}{\Delta t} \phi_P^o + \beta S_{U_P}^n V_P^n + \beta \sum_{f=1}^m \underbrace{\Gamma_f^n \mathbf{k}_f \cdot (\nabla \phi)_f^n}_{\text{non-orthogonal}} - (1-\beta) \sum_{f=1}^m \rho_f^o F_f^o \phi_f^o \\ &+ (1-\beta) \sum_{f=1}^m \Gamma_f^o \mathbf{S}_f \cdot (\nabla \phi)_f^o + (1-\beta) S_{P_P}^o V_P^o \phi_P^o + (1-\beta) S_{U_P}^o V_P^o \end{aligned} \quad (4.33)$$

After having employed the aforementioned discretisation and linearisation procedures, we arrive to equation (4.33). One important detail in the above re-arrangement is the

separation of the diffusion term of the new time level into orthogonal and non-orthogonal contributions; the orthogonal term, as it has already been mentioned can be linearised and it is located on the l.h.s. of equation (4.33), whereas the non-orthogonal contribution has been moved to the r.h.s. of the equation. Although obvious, it should be mentioned that all terms on the l.h.s. of equation (4.33) are linear functions of the unknown  $\varphi_j^n$ , where with  $j$  the cell index of any control volume is denoted. On the r.h.s. of the above equation all the terms have been grouped, which refer to either the previous time level or to the current one, but could not be linearised. Equation (4.33) is rewritten in a generic form that resembles a linear system:

$$A_P \varphi_P^n = \sum_N A_N \varphi_N^n + B_P \quad (4.34)$$

After having formulated the linear system, the iterative solution procedure that is followed to obtain the final solution at the new time level is described. Effectively, due to the linearisation of the original mathematical equations, the coefficients of both  $A$  and  $B$  matrices depend on the targeted solution (although for some problems it is possible that matrix  $A$  is independent). For this reason an iterative procedure is undertaken. At the beginning of each time step, the solution from the previous one is used (or an initial guess, if the first time step is being solved), in order to evaluate matrices  $A$  and  $B$ . Subsequently the system described by equation (4.34) is solved numerically with an iterative solver, and then at the end of the iteration all parameters in these matrices that depend on the solution of the current time step are updated. This cyclic procedure goes on until a converged solution for  $\varphi$  is reached. Because of the intrinsic non-linear nature of the system of equations, it is common practice to introduce an artificial modification to the linear system, which is termed *under-relaxation*:

$$A_P \varphi_P^n + \frac{1-\zeta}{\zeta} A_P \varphi_P^n = \sum_N A_N \varphi_N^n + B_P + \frac{1-\zeta}{\zeta} A_P \varphi_P^o \quad (4.35)$$

This leads to this final form:

$$\frac{A_P}{\zeta} \varphi_P^n = \sum_N A_N \varphi_N^n + B_P + \frac{1-\zeta}{\zeta} A_P \varphi_P^o \quad (4.36)$$

where  $\zeta$  is the under-relaxation factor ( $0 \leq \zeta \leq 1$ ). With this modification the convergence procedure is enhanced, due to the improved diagonal dominance of the modified matrix  $A$ . The final detail about the method is the choice of the algebraic system solver. The solution of the modified linear system can be obtained with either a direct solver or an iterative one; summaries on both approaches can be found in [204]. Due to the CPU

cost of direct solvers most CFD codes employ iterative ones. The GFS solver features the extensive SLAP library [210] of iterative solvers; from various tests within the research group it has been concluded that the Bi-Conjugate Gradient Sparse Matrix solver, which utilises iterative preconditioning, is efficient enough for the flow problems investigated here. The aforementioned solver is an extension of the conjugate gradient method [55] that is valid also for non-symmetric linear systems but at twice the cost per iteration. More details about the solver and its limitations are presented in [211]. This solver has been contributed to the SLAP library by Greenbaum and Seager [210] and it is used throughout this study. This final step concludes the description of the framework and the specific details of the finite volume method.

## 4.5 DISCRETISATION OF SINGLE PHASE MODEL EQUATIONS

### 4.5.1 Continuity Equation

The continuity equation (3.8) is integrated over an arbitrary control volume and discretised; this leads to the following expression:

$$\int_{V_p} \nabla \cdot \mathbf{u} dV = 0 \Rightarrow \sum_f F_f^n = 0 \quad (4.37)$$

where it is reminded that  $F_f = \mathbf{S}_f \cdot \mathbf{u}_f$  and that with  $n$  the new time level is denoted. As it was discussed in Section 4.4.1, a special interpolation technique for the velocity face values of the transport coefficients is necessary, whose details will be given in Section 4.5.2.

### 4.5.2 Momentum Equation

The momentum equation is the most complex among the transport equations; its discretisation is the lengthiest and features important intrinsic details. We start with the semi-discretised form of the momentum equation (4.38):

$$\begin{aligned} \int_t^{t+\Delta t} \left[ \int_{V_p} \frac{\partial(\rho \mathbf{u})}{\partial t} dV + \int_{V_p} \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV \right] dt = & - \int_t^{t+\Delta t} \left[ \int_{V_p} \nabla p dV \right] dt \\ & + \int_t^{t+\Delta t} \left[ \int_{V_p} \nabla \cdot (\mu_e (\nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T)) dV + \int_{V_p} \rho \mathbf{g} dV \right] dt \end{aligned} \quad (4.39)$$

where  $\mu_e$  is the effective viscosity, which represents the combined effect of viscous and turbulent momentum dissipation ( $\mu_e = \mu + \mu_t$ ). Each term within the time integrals in the above equation is discretised following the finite volume method:

$$\begin{aligned}
\int_t^{t+\Delta t} \left[ \frac{\rho(\mathbf{u}V)_P^{t+\Delta t} - \rho(\mathbf{u}V)_P^t}{\Delta t} + \sum_f \rho F_f \mathbf{u}_f \right] dt = & - \int_t^{t+\Delta t} [(\nabla p)_P V_P] dt \\
& + \int_t^{t+\Delta t} \left[ \sum_f \mu_{ef} \mathbf{S}_f \cdot (\nabla \otimes \mathbf{u})_f + (\nabla \otimes \mathbf{u})_P (\nabla \mu_e)_P V_P + \rho \mathbf{g} V_P \right] dt
\end{aligned} \quad (4.40)$$

An important detail about the discretisation of the momentum equation is that the face velocities  $\mathbf{u}_f$  of the convection term are estimated with the Gamma interpolation scheme of Jasak et al [207], which was discussed in Section 4.4.1. The second term on the r.h.s. of equation (4.40) is the part of the viscous tensor that can be discretised like the diffusion term in Section 4.4.5; this means that it can be divided into orthogonal and non-orthogonal contributions, which lead to:

$$\begin{aligned}
& \int_t^{t+\Delta t} \left[ \frac{\rho(\mathbf{u}V)_P^{t+\Delta t} - \rho(\mathbf{u}V)_P^t}{\Delta t} + \sum_f \rho F_f \mathbf{u}_f \right] dt - \\
& \int_t^{t+\Delta t} \left[ \sum_f \mu_{ef} \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f \right] dt = - \int_t^{t+\Delta t} [(\nabla p)_P V_P] dt \\
& + \int_t^{t+\Delta t} \left[ \sum_f \mu_{ef} \mathbf{k}_f \cdot (\nabla \otimes \mathbf{u})_f + (\nabla \otimes \mathbf{u})_P (\nabla \mu_e)_P V_P + \rho \mathbf{g} V_P \right] dt
\end{aligned} \quad (4.41)$$

At this point, the same assumption about the time level approximation that was discussed in Section 4.4.3 is introduced. Following equation (4.33), the blending coefficient  $\beta$  is introduced;  $\beta = 1$  corresponds to the Euler implicit scheme, and  $\beta = 0.5$  corresponds to the Crank-Nicolson scheme:

$$\frac{\rho V_P^n}{\Delta t} \mathbf{u}_P^n + \beta \sum_f \rho F_f^n \mathbf{u}_f^n - \beta \sum_f \mu_{ef}^n \mathbf{D}_f \cdot (\nabla \otimes \mathbf{u})_f^n = S_{U_P} - \beta (\nabla p)_P^n V_P^n \quad (4.42)$$

where the source term  $S_U$  contains all the remaining terms:

$$\begin{aligned}
S_{U_P} = & \frac{\rho V_P^o}{\Delta t} \mathbf{u}_P^o + \beta \sum_f \mu_{ef}^n \mathbf{k}_f \cdot (\nabla \otimes \mathbf{u})_f^n + \beta (\nabla \otimes \mathbf{u})_P^n (\nabla \mu_e)_P^n V_P^n \\
& + \beta \rho \mathbf{g} V_P^n - (1-\beta) (\nabla p)_P^o V_P^o - (1-\beta) \sum_f \rho F_f^o \mathbf{u}_f^o + (1-\beta) \rho \mathbf{g} V_P^o \\
& + (1-\beta) (\nabla \otimes \mathbf{u})_P^o (\nabla \mu_e)_P^o V_P^o + (1-\beta) \sum_f \mu_{ef}^o \mathbf{S}_f \cdot (\nabla \otimes \mathbf{u})_f^o
\end{aligned} \quad (4.43)$$

On the r.h.s. of equation (4.42) all terms refer either to the previous time step (o) or to the current one (n), but cannot be linearised for the unknown velocity  $\mathbf{u}_P^n$ , and thus have to be treated explicitly; within the context of the iterative solution algorithm, this effectively means that the solution of the previous iteration is used for their evaluation. From

equations (4.42) and (4.43) the algebraic system for the velocity vector in the new time level is formulated:

$$A_P \mathbf{u}_P^n = \sum_N A_N \mathbf{u}_N^n + S_{U_P} - \beta (\nabla p)_P^n V_P^n \quad (4.44)$$

The pressure gradient has been left out from the r.h.s. source term on purpose, due to its importance. Moreover, it is useful for the forthcoming discussion on the pressure correction method to represent the discretised momentum equation in the following form:

$$\mathbf{u}_P^n = \frac{\mathbf{H}(\mathbf{u})_P}{A_{P_P}} - \frac{\beta V_P^n}{A_{P_P}} (\nabla p)_P^n \quad (4.45)$$

where  $A_{P_P} \equiv A_P$  and:

$$\mathbf{H}(\mathbf{u})_P = \sum_N A_N \mathbf{u}_N^n + S_{U_P} \quad (4.46)$$

This concludes the discretisation of the momentum equation for single-phase flows.

### Turbulence Model Equations

The discretisation of equations (3.13) and (3.16) is quite straightforward. For the equation of the turbulent kinetic energy  $k$  we have:

$$\frac{\rho V_P^n}{\Delta t} k_P^n + \beta \sum_f \rho F_f^n k_f^n - \beta \sum_f \left( \mu + \frac{\mu_t}{\sigma_k} \right)_f^n \mathbf{D}_f \cdot (\nabla k)_f^n = S_{U_P} \quad (4.47)$$

where

$$\begin{aligned} S_{U_P} = & \frac{\rho V_P^o}{\Delta t} k_P^o + \beta \sum_f \left( \mu + \frac{\mu_t}{\sigma_k} \right)_f^n \mathbf{k}_f \cdot (\nabla k)_f^n + \beta P_P^n V_P^n - \beta \rho \varepsilon_P^n V_P^n \\ & - (1 - \beta) \sum_f \rho F_f^o k_f^o + (1 - \beta) \sum_f \left( \mu + \frac{\mu_t}{\sigma_k} \right)_f^o \mathbf{S}_f \cdot (\nabla k)_f^o \\ & + (1 - \beta) P_P^o V_P^o - (1 - \beta) \rho \varepsilon_P^o V_P^o \end{aligned} \quad (4.48)$$

The above expression constitutes the linear system that needs to be solved. Similarly for the equation of turbulence dissipation  $\varepsilon$  we derive the following expression:

$$\frac{\rho V_P^n}{\Delta t} \varepsilon_P^n + \beta \sum_f \rho F_f^n \varepsilon_f^n - \beta \sum_f \left( \mu + \frac{\mu_t}{\sigma_k} \right)_f^n \mathbf{D}_f \cdot (\nabla \varepsilon)_f^n = S_{P_P} \varepsilon_P^n + S_{U_P} \quad (4.49)$$

where

$$\begin{aligned}
S_{U_P} = & \frac{\rho V_P^o}{\Delta t} \varepsilon_P^o + \beta \sum_f \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right)_f^n \mathbf{k}_f \cdot (\nabla \varepsilon)_f^n + \beta \frac{\varepsilon_P^n}{k_P^n} C_{\varepsilon 1}^n P_P^n V_P^n \\
& - (1 - \beta) \sum_f \rho F_f^o \varepsilon_f^o + (1 - \beta) \sum_f \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right)_f^o \mathbf{S}_f \cdot (\nabla \varepsilon)_f^o \\
& + (1 - \beta) \frac{\varepsilon_P^o}{k_P^o} C_{\varepsilon 1}^o P_P^o V_P^o - (1 - \beta) \frac{\varepsilon_P^o}{k_P^o} C_{\varepsilon 2}^o \varepsilon_P^o V_P^o
\end{aligned} \tag{4.50}$$

and

$$S_{P_P} = -\beta \frac{\varepsilon_P^n}{k_P^n} C_{\varepsilon 2}^n V_P^n \tag{4.51}$$

It is clear from the above equations that the source term of equation (3.16) was linearised, in order to have a positive contribution to the linear system. With this linearisation the final form of the linear system for  $\varepsilon_P^n$  is:

$$(A_P - S_{P_P}) \varepsilon_P^n = \sum_N A_N \varepsilon_N^n + S_{U_P} \tag{4.52}$$

In concluding, the coefficients  $C_{\varepsilon 1}$  and  $C_{\varepsilon 2}$  in the above equations have been considered as time dependent, to include the case of the non-equilibrium  $k$ - $\varepsilon$  model.

### 4.5.3 Pressure Equation

One major problem with modelling incompressible flows is the lack of a constitutive equation for pressure; to circumvent this shortcoming many alternative methods have been proposed in the literature over the recent years [204]. In the current study, as it has been already discussed, the pressure-correction method is employed [212]. Within the context of this method an equation for the pressure is derived, by combining the discretised continuity and momentum equations. Before describing this procedure, it is useful to represent the aforementioned discretised continuity equations in a unified way for completeness:

$$\sum_f q_f \mathbf{S}_f \cdot \mathbf{u}_f^n = S_C \tag{4.53}$$

where the terms  $q$  and  $S_C$  will be assumed  $q = 1$  and  $S_C = 0$  for the rest of the analysis since only Single-Phase flow is considered. For the extension of the method for Eulerian Mixture and Eulerian – Lagrangian the reader can refer to [18].

It is necessary at this point to obtain the interpolated velocity  $\mathbf{u}_f^n$  at the cell faces. Nevertheless, since the control volume topology is collocated and because these face veloc-

ity values are coupled with pressure, through the pressure correction method, the interpolation approaches presented in Section 4.4.1 cannot be employed for this purpose. It has been established from past studies that usage of such techniques would cause the well-known *check-board pressure* effect [213], which effectively means that a partial decoupling between pressure and velocity occurs.

A special interpolation method needs to be employed, namely the approach of Rhie and Chow [214]. With this technique the discretised momentum equation is used to predict the face value of the velocity; this is done by isolating the contribution of the pressure gradient from equation (4.45), when interpolating it to the face. The contribution of the pressure gradient at the face is then added explicitly to the interpolated face velocity, by calculating it directly from the pressure values of the two cell nodes, which are adjacent to the face. In equation form the above procedure reads:

$$\mathbf{u}_f^n = \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f - \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla P)_f^n \quad (4.54)$$

where, the face values other than the pressure gradient are calculated with central differencing; recalling equation (4.55) we have:

$$\left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f = f_P \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_P + (1 - f_P) \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_N \quad (4.56)$$

and

$$\left( \frac{\beta V_P^n}{A_P} \right)_f = f_P \left( \frac{\beta V_P^n}{A_P} \right)_P + (1 - f_P) \left( \frac{\beta V_P^n}{A_P} \right)_N \quad (4.57)$$

Substitution of  $\mathbf{u}_f^n$  from equation (4.54) into (4.53) equation gives:

$$\sum_f q_f \mathbf{S}_f \cdot \left[ \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f - \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla P)_f^n \right] = S_{C_P} \quad (4.58)$$

or

$$\sum_f q_f \mathbf{S}_f \cdot \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla P)_f^n = \sum_f q_f \mathbf{S}_f \cdot \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f - S_{C_P} \quad (4.59)$$

The term containing the pressure gradient over the faces is calculated in a similar way to the diffusion term of the generic transport equation, which means that it can be decomposed into orthogonal and non-orthogonal components. Recalling Section 4.4.2, where the calculation of gradients with the above decomposition was discussed, we have:

$$\mathbf{S}_f \cdot (\nabla p)_f^n = \underbrace{\mathbf{D}_f \cdot (\nabla p)_f^n}_{\text{orthogonal}} + \underbrace{\mathbf{k}_f \cdot (\nabla p)_f^n}_{\text{non-orthogonal}} \quad (4.60)$$

The orthogonal contribution is treated implicitly and is given by:

$$\mathbf{D}_f \cdot (\nabla p)_f^n = |\mathbf{D}_f| \frac{p_N - p_P}{|\mathbf{d}_f|} \quad (4.61)$$

The non-orthogonal contribution is treated explicitly, which means that pressure values obtained from the previous solution iteration are utilised. Furthermore, for this term the pressure gradient  $(\nabla p)_f^n$  at the cell face is obtained by central differencing interpolation, from the pressure gradient values at the centres of the two adjacent cells. All the components for the derivation of the pressure equation are available then; the final form of this equation reads:

$$\sum_f q_f \mathbf{D}_f \cdot \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla p)_f^n = S_{P_P} \quad (4.62)$$

where

$$S_{P_P} = \sum_f q_f \mathbf{S}_f \cdot \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f - S_{C_P} - \sum_f q_f \mathbf{k}_f \cdot \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla p)_f^n \quad (4.63)$$

Equation (4.63) constitutes the linear system for pressure that needs to be solved; it can be reformulated in the following form:

$$A_P p_P = \sum_N A_N p_N + S_{P_P} \quad (4.64)$$

After the solution of equation (4.64), the newly obtained pressure field is used to update the volumetric fluxes at the cell faces with equation (4.45):

$$F_f^n = \mathbf{S}_f \cdot \left( \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_f - \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla p)_f^n \right) \quad (4.65)$$

With the derivation of the pressure equation the discretisation of the governing equations for the interior of the flow domain is concluded. Nevertheless, the treatment of the boundary conditions for these equations needs to be discussed; this is done in the following paragraph.

#### 4.5.4 Boundary Conditions

In Section 4.4.7 the application of numerical boundary conditions to the generic transport equation was discussed, with the coverage of fixed value and fixed gradient boundary conditions. Nevertheless, this discussion needs to be extended here, since some of



the aforementioned model equations require special treatment with respect to their boundary conditions. Although the application of fixed value and fixed gradient boundary conditions for the equation of the above models follows the approach of Section 4.4.7, some conditions of these types need some attention and will be covered initially. Subsequently, the aforementioned required extension for the boundary conditions of some variables will be discussed, and then a general overview of the combinations of boundary conditions which are applied for some typical problems will be given.

### Fixed Value Boundary Conditions

The boundary control volume that was depicted in Figure 4.2 will be used as reference for the following discussion. The prescription of a fixed pressure value at a boundary face  $f$  needs some attention, since with this condition the boundary face velocities have to be calculated from equation (4.62). For this particular case, the term

$$\mathbf{D}_f \cdot \left( \frac{\beta V_P^n}{A_P} \right)_f (\nabla p)_f^n \quad (4.66)$$

needs to be calculated at the boundary face. Since the term  $V_P^n/A_P$  does not exist at this location, the necessary face value of this term will be approximated with the value of the internal boundary cell. The fixed boundary value  $p_b$  is used for the approximation of the face pressure gradient  $(\nabla p)_f^n$ . The implementation of the above assumptions in equation (4.62) leads to the following:

$$|\mathbf{D}_f| \cdot \left( \frac{\beta V_P^n}{A_P} \right)_P \frac{p_b - p_P}{|\mathbf{d}_f|} = \mathbf{S}_f \cdot \left( \frac{\mathbf{H}(\mathbf{u})}{A_P} \right)_P \quad (4.67)$$

At the boundary faces the source  $S_C$  of equation (4.53) is assumed to be zero. For the remaining governing equations the implementation of fixed value boundary conditions does not impose any particular difficulty; their diffusion and convection terms at the boundary faces are calculated according to the approach described in Section 4.4.7.

### Fixed Gradient Boundary Conditions

A boundary condition of this type that needs to be mentioned is the condition for the pressure equation at solid boundaries. Typically, for the velocity a zero-slip condition is applied, and for the turbulent kinetic energy  $k$  and its dissipation rate  $\varepsilon$  a special treatment is needed; for all other remaining variables, except the pressure, a zero gradient condition is imposed at faces which are adjacent to solid boundaries. By the case of pressure though, there is the special requirement on the imposed boundary condition; it

must comply with the hydrostatic phenomenon. For this reason, at solid boundaries, the following boundary condition is implemented for the pressure equation:

$$(\nabla p)_f^n = \rho \mathbf{g} \quad (4.68)$$

which leads to:

$$p_b = p_P + \rho(\mathbf{d}_n \cdot \mathbf{g}) \quad (4.69)$$

As for the case of fixed value boundary conditions, the implementation of fixed gradient boundary conditions for the remaining variables is done according to the approach that was described in Section 4.4.7.

### Near-Wall Turbulence Treatment

For the purposes of the current study the so-called *wall function* approach of Launder and Spalding [200] has been utilised. The wall function approach relies on the existence of a two-layer structure within the turbulent boundary layer. The velocity profile within these two layers can be evaluated from the following:

$$u^+ = \frac{\bar{u}_t}{u_\tau} = \begin{cases} y^+ & y^+ \leq 11.63 \\ \frac{1}{\kappa} \ln(Ey^+) & y^+ > 11.63 \end{cases} \quad (4.70)$$

where  $\bar{u}_t$  is the mean velocity parallel to the wall,  $u_\tau$  is the shear velocity,  $u_\tau = \sqrt{\tau_w/\rho}$ ,  $\tau_w$  is the shear stress at the wall,  $\kappa$  is the so-called von Karman constant ( $\kappa=0.41$ ),  $E$  is an empirical constant that depends on the wall roughness and is related to the thickness of the viscous sub layer ( $E = 9.7$  for smooth walls) and  $y^+$  is the dimensionless distance from the wall:

$$y^+ = \frac{\rho u_\tau y}{\mu} \quad (4.71)$$

But from [204]:

$$u_\tau = C_\mu^{1/4} \sqrt{k} \quad (4.72)$$

where  $C_\mu$  is given in Table 3.1. By combining equations (4.71) and (4.72) we have:

$$y^+ = \frac{\rho C_\mu^{1/4} \sqrt{k} y}{\mu} \quad (4.73)$$

and

$$\tau_w = \rho u_\tau^2 = \rho C_\mu^{1/4} \kappa \sqrt{k} \frac{\bar{u}_t}{\ln(y^+ E)} \quad (4.74)$$

Equation (4.74) is utilised in vector form, in order to calculate the additional wall shear force due to turbulence; this force is added as an extra source term then to the discretised momentum equation of the boundary control volumes.

### Outline of Boundary Conditions

Depending on the flow problem that needs to be simulated, a combination of boundary conditions for all the aforementioned equations has to be employed. In Table 4.1 an overview of these boundary conditions combinations is given.

Variable	Wall	Inlet	Outlet	Symmetry Plane
$U$	Fixed Value (No-slip)	Fixed Value Continuity <sup>2</sup>	Zero Gradient Continuity <sup>1</sup>	Zero Gradient
$P$	Fixed Gradient	Zero Gradient Fixed Value	Fixed Value	Zero Gradient
$K$	Fixed Value	Fixed Value	Zero Gradient	Zero Gradient
$E$	Fixed Value	Fixed Value	Zero Gradient	Zero Gradient

**Table 4.1: Summary of boundary conditions for each equation and boundary type**

This concludes the discussion on the issues that are pertinent to the physics and the implementation of boundary conditions to each equation.

### 4.5.5 Sequence of Solution

The approach that is followed for solving the coupled system of discretised equations is the so-called *segregated* approach. Within the context of this approach, the GFS solver performs a number of iterations, aiming to satisfy simultaneously the above-presented set of differential equations on a sequential basis through numerical convergence. In Section 4.5.2, where the pressure equation was derived, it was shown that pressure and velocity have strong linear coupling. This coupling is employed by the pressure-correction method, which is an intrinsic part of the current iterative solution algorithm. Many variants of the pressure-correction method exist [204], with the most well known being the SIMPLE [212] and the PISO [215] algorithms. The GFS solver features both

<sup>2</sup> When a constant pressure inlet or outlet is prescribed the boundary face velocity is calculated from the continuity equation.

options; nevertheless, throughout this study the PISO algorithm has been used for the flow equations and the SIMPLE for the adjoint. The latter is a pressure-velocity method specially developed for non-iterative computation of unsteady compressible flows, but is easily adaptable to incompressible ones.

#### 4.5.6 Pressure-velocity coupling with PISO

The basic steps of the PISO algorithm can be described as follows:

- **Momentum prediction:** The discretised momentum equation (4.44) is solved initially with an approximate pressure field  $p^*$ ; this field corresponds to that of the previous iteration of the current time step or of the last iteration of the previous time step. The solution of the momentum equation gives a new velocity field  $\mathbf{u}^*$  that does not satisfy the continuity equation. It goes without saying that the momentum equation (4.44) is depicted in linear system form; since with this formulation any of the above discretised momentum equation can be represented.
- **Pressure solution:** This predicted velocity  $\mathbf{u}^*$  is used to calculate  $\mathbf{H}(\mathbf{u}^*)$  from equation (4.46); the latter is used for the calculation of a new pressure field  $p^{**}$  from the pressure equation (4.64). If the mesh is non-orthogonal then the source term of equation (4.63) is updated and a more accurate solution is obtained.
- **Explicit velocity correction:** From equation (4.65) a new set of conservative volumetric fluxes (also referred to as *transport coefficients*) is obtained, which is consistent with the new pressure field. The latter is used in equation (4.45) in order to perform an explicit correction on velocity. The newly calculated velocity field  $\mathbf{u}^{**}$  is now consistent with the new pressure field. With this velocity field  $\mathbf{H}(\mathbf{u}^{**})$  is calculated again from equation (4.46) and is used one more time to obtain a new pressure  $p^{***}$ , by solving the pressure equation (4.64). The iteration over the last two steps is continued until a pre-defined numerical tolerance has been met.

#### 4.5.7 Solution procedure of the governing equations system

In the previous paragraphs the discretisation of all the governing conservation equations was addressed; moreover, the implementation of the necessary boundary conditions to these equations, together with the pressure-velocity coupling through the PISO pressure-correction method, was covered as well. It is now possible to describe the solution

sequence of the aforementioned theoretical models. In Table 4.2 the sequence of the computational steps that are followed is provided.

1. Initialise all variables.
2. Solve the momentum equation (4.75).
3. Perform PISO-Loop.
4. Solve  $k$ - $\epsilon$  equations (according to which model is employed).  
For the non-equilibrium  $k$ - $\epsilon$  only:  
Update  $C_{\epsilon 1}$  and  $C_{\epsilon 2}$  coefficients from Table 3.1
5. If convergence residuals are above the required numerical tolerance, repeat steps 2-6 again.
6. If the desired simulation time has not been reached yet advance to the next time step and return to step 2.

**Table 4.2: Numerical solution procedure**

## 4.6 SOLUTION OF THE ADJOINT SYSTEM OF EQUATIONS

The system of adjoint equations is very similar to the system of flow equations so the procedure described in chapter 4 for the discretisation and solution of the Navier-Stokes equations is followed here as well. The same methods are employed for the discretisation and interpolation of the terms of the corresponding costate momentum equation. For  $\Psi_p$  an equivalent pressure correction method is employed. There are of course some major differences in the two systems which are stated in this section one by one while proceeding with the analysis.

### 4.6.1 Adjoint ‘Continuity’ Equation

As in the case of the continuity equation, the first equation of the adjoint system (3.48) is integrated over an arbitrary control volume and discretised; this leads to the following expression:

$$\int_{V_p} \nabla \cdot \Psi_u dV = 0 \Rightarrow \sum_f (F_{adj})_f = 0 \quad (4.76)$$

where it is reminded that  $(F_{adj})_f = S_f \cdot (\Psi_u)_f$  and there is no time discretisation because in this study the adjoint equations are solved for steady state.

### 4.6.2 Adjoint ‘Momentum’ Equation

Unlike the adjoint of the continuity equation the adjoint of the momentum equation is slightly different and need to be treated differently than the actual momentum. The differences are stated below.

#### Convection terms

The convection term of the second equation of the adjoint system (3.48) is very similar to the convection term of the momentum equation though a minus sign appear in the case of the adjoint equations. The convection terms generally indicate transport so the minus sign means that information transfer takes place in the opposite direction of the information propagation of the costate variables in relation to the flow variables. To overcome this we can use a downwind scheme for the interpolation of the adjoint variables. Generally downwind schemes result in instabilities in the solution so in this study the preferred approach is to solve using the same interpolation schemes as for the flow solver (upwind) and solve for  $-\Psi_u$ .

Another difference between the adjoint and flow convection term is that the adjoint is linear because there are no costate variables appearing in the volumetric flux. This is the most important difference of the two equations and is the reason that the adjoint equations converge much faster. During the adjoint solution the already calculated values of the velocities are replaced in the volumetric flux and there is no need for special treatment as in the case of the calculation of the velocities in section 4.5.3.

#### Diffusion terms

The diffusion terms appearing in the adjoint equations (3.48) have exactly the same form as in the momentum equation (3.9). In the adjoint case the effective viscosity  $\mu_e$  is already known from the solution of the turbulent equations and as it has been mentioned in section 3.8.4 it is kept frozen during the adjoint solution.

The diffusion terms are re-arranged into orthogonal and non-orthogonal contributions; the orthogonal term can be linearised and it is located on the l.h.s. of equation, whereas the non-orthogonal contribution is moved to the r.h.s. of the equation (3.46). Although obvious, it should be mentioned that all terms on the l.h.s. of equation (3.48) are linear-

ised functions of the unknown  $\Psi$ . On the r.h.s. of equation (3.48) exist the terms which cannot be linearised. Apart from the diffusion term there is also a convection term which cannot be linearised and has been moved to the r.h.s. Also the  $\Psi_p$  gradient has been left out from the r.h.s. source term on purpose, as in the flow solver case, due to its importance.

Now that the differences are stated the finite volume method is employed in equation (3.48) as for the flow solver to conclude:

$$-\sum_f \rho (F_{adj})_f (\Psi_u)_f - \sum_f \mu_{ef} \mathbf{D}_f \cdot (\nabla \otimes \Psi_u)_f =$$

$$(\nabla \Psi_p)_p V_P + \sum_f \mu_{ef} \mathbf{k}_f \cdot (\nabla \otimes \Psi_u)_f + (\nabla \otimes \Psi_u)_p (\nabla \mu_e)_p V_P + \rho \mathbf{u}_p (\nabla \otimes \Psi_u)_p^T V_P \quad (4.77)$$

From the above equations the following algebraic system is formulated:

$$A_P \mathbf{u}_P = \sum_N A_N \mathbf{u}_N + S_{\Psi_p} - (\nabla \Psi_p)_p V_P \quad (4.78)$$

Where 
$$S_{\Psi_p} = \sum_f \mu_{ef} \mathbf{k}_f \cdot (\nabla \otimes \Psi_u)_f + (\nabla \otimes \Psi_u)_p (\nabla \mu_e)_p V_P + \rho \mathbf{u}_p (\nabla \otimes \Psi_u)_p^T V_P$$

For the solution of the algebraic systems, of equation (4.77), the SLAP library [210] is used with the Bi-Conjugate Gradient Sparse Matrix solver for non-symmetric linear systems, which utilises iterative preconditioning, used as the most efficient one.

#### 4.6.3 Pressure correction for $\Psi_p$

In the adjoint equation case again the major problem for the solution of  $\Psi_p$  is the lack of a constitutive equation for  $\Psi_p$ ; to circumvent this shortcoming in the current study, the pressure-correction method is employed [212]. Within the context of this method an equation for  $\Psi_p$  is derived, by combining the discretised continuity and momentum equations the same way as for the pressure.

Using the same special interpolation that has been used for the pressure the contribution of the  $\Psi_p$  gradient at the face is then added explicitly to the interpolated face velocity, by calculating it directly from the  $\Psi_p$  values of the two cell nodes, which are adjacent to the face. In equation form the above procedure reads:

$$(\Psi_u)_f = \left( \frac{\mathbf{H}(\Psi_u)}{A_P} \right)_f - \left( \frac{V_P}{A_P} \right)_f (\nabla \Psi_p)_f \quad (4.79)$$

where the face values other than the pressure gradient are calculated with central differencing.

Substitution of  $(\Psi_u)_f$  from equation (4.79) into equation the adjoint continuity equation (3.48) gives:

$$\sum_f \mathbf{S}_f \cdot \left( \left( \frac{\mathbf{H}(\Psi_u)}{A_P} \right)_f - \left( \frac{V_P}{A_P} \right)_f (\nabla \Psi_p)_f \right) = 0 \quad (4.80)$$

or

$$\sum_f \mathbf{S}_f \cdot \left( \frac{V_P}{A_P} \right)_f (\nabla \Psi_p)_f = \sum_f \mathbf{S}_f \cdot \left( \frac{\mathbf{H}(\Psi_u)}{A_P} \right)_f \quad (4.81)$$

The term containing the  $\Psi_p$  gradient over the faces is calculated in a similar way to the diffusion term of the generic transport equation, which means that it can be decomposed into orthogonal and non-orthogonal components. Recalling Section 4.4.2, where the calculation of gradients with the above decomposition was discussed, we have:

$$\mathbf{S}_f \cdot (\nabla \Psi_p)_f = \underbrace{\mathbf{D}_f \cdot (\nabla \Psi_p)_f}_{\text{orthogonal}} + \underbrace{\mathbf{k}_f \cdot (\nabla \Psi_p)_f}_{\text{non-orthogonal}} \quad (4.82)$$

The orthogonal contribution is treated implicitly and is given by:

$$\mathbf{D}_f \cdot (\nabla \Psi_p)_f = |\mathbf{D}_f| \frac{(\Psi_p)_N - (\Psi_p)_P}{|\mathbf{d}_f|} \quad (4.83)$$

The non-orthogonal contribution is treated explicitly, which means that  $\Psi_p$  values obtained from the previous solution iteration are utilised. Furthermore, for this term the pressure gradient  $(\nabla \Psi_p)_f$  at the cell face is obtained by central differencing interpolation, from the  $\Psi_p$  gradient values at the centres of the two adjacent cells. All the components for the derivation of the pressure equation are available then; the final form of this equation reads:

$$\sum_f \mathbf{D}_f \cdot \left( \frac{V_P}{A_P} \right)_f (\nabla \Psi_p)_f = S_{P_p} \quad (4.84)$$

where



$$S_{P_p} = \sum_f \mathbf{S}_f \cdot \left( \frac{\mathbf{H}(\Psi_u)}{A_p} \right)_f - \sum_f \mathbf{k}_f \cdot \left( \frac{V_p}{A_p} \right)_f (\nabla \Psi_p)_f \quad (4.85)$$

Equation (4.84) constitutes the linear system for  $\Psi_p$  that needs to be solved; it can be reformulated in the following form:

$$A_p(\Psi_p)_p = \sum_N A_N(\Psi_p)_N + S_{P_p} \quad (4.86)$$

Due to the linearity of the convection terms no special treatment is required for the calculation of the volumetric flux which is already calculated during the flow solution. The GFS solver features SIMPLE and PISO pressure correction methods. Throughout this study the PISO algorithm has been used for the flow equations but for the adjoint the SIMPLE algorithm has been preferred due to the simplicity of the equations. Since the adjoint equations are linear there is no need to use a more complicated algorithm like PISO.

#### 4.6.4 Boundary conditions

The boundary conditions of the adjoint equations are treated in the same way as the ones of the flow solver. In the case of the target pressure distribution cost function the r.h.s of eq. (3.51) is calculated in each optimisation cycle using the current pressure values. Then the calculated r.h.s is enforced for the values of  $\Psi_u$  on the wall ‘pseudo-cells’. The same procedure is used for the given  $C_d$  case. The necessary quantities are calculated using the known velocities from the flow field and then they are enforced for the values of  $\Psi_u$  on the exit ‘pseudo-cells’.

At this point all the previously discretised parts of the adjoint equation can be assembled, in order to form the final set of algebraic equations that need to be solved. The solution procedure is exactly the same as for the flow solver and it has already been described in section 4.5.7.

### 4.7 SUMMARY

In this chapter the flow solution procedure has been described and cross referenced with the solution of the adjoint equations. The control volume approach has been adopted for the analysis of the fluid flow properties and subsequently for the adjoint field properties as well. Practically, the adjoint equations were formed in a similar way as the flow

equations so the same methods were used for their solution. The discretisation of the generic equation has been used as the origin for the discretisation and solution of both the flow and adjoint equations. Basic differences of the two sets of equations are stated and the different treatments are discussed. The following chapters are dedicated to the validation and application of the optimisation method described in this thesis.

## *Chapter 5*

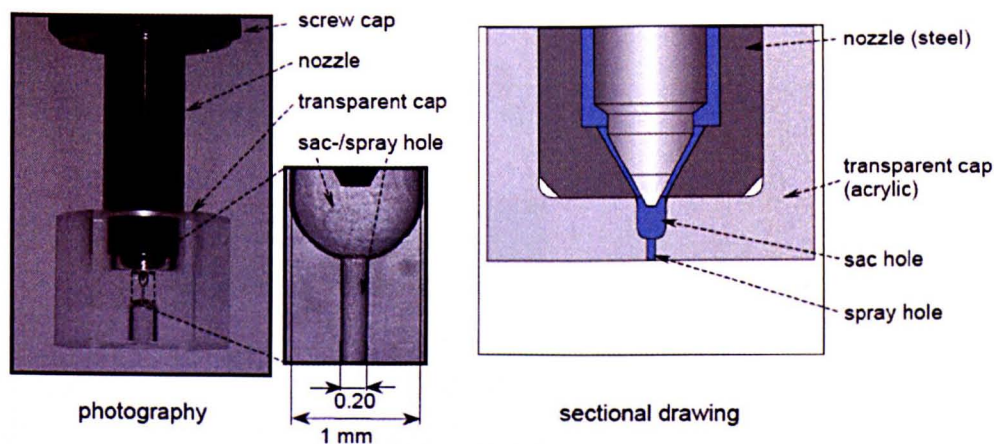
# *METHOD VALIDATION & PARAMETRIC STUDIES*

### **5.1 INTRODUCTION**

In this chapter the validation of the proposed method, as this was described in chapter 3, is discussed in detail. The effect of various numerical parameters of the flow and the adjoint equation as well as parameters related to the optimisation method on the overall optimisation procedure is explored. With respect to the numerical issues, a number of parametric studies are presented, through which the effect of the spatial (grid) discretisation and the convergence of the flow and adjoint equations is assessed. Concerning the optimisation parameters, study on the most suitable step sizes of the steepest descent method is discussed. For the validation of the model the inverse design with respect to a prescribed pressure distribution of a 2-D axisymmetric nozzle, using different targets and different initial geometries, is presented. Results of the inverse design of a 3-D VCO multi-hole nozzle are also presented. In order to show the generality of the proposed method a 2-D airfoil inverse design case is presented here as well.

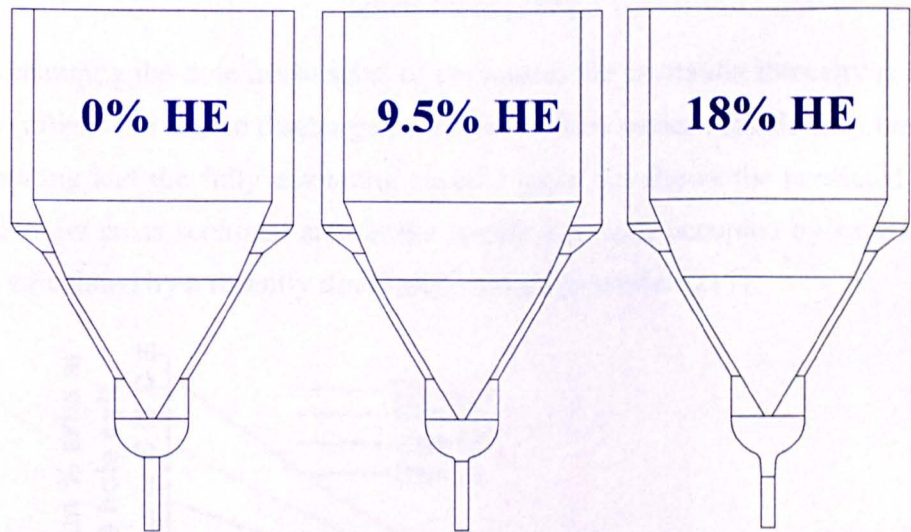
### **5.2 REFERENCE NOZZLE GEOMETRY FOR PARAMETRIC STUDIES**

Within the context of the I-Level European Project several real-size single-hole nozzles were manufactured [180] for the experimental investigation of cavitation under realistic flow conditions. These nozzles were operated under a variety of injection and back pressures. An axisymmetric single-hole nozzle that was made from a transparent acrylic material (Figure 5.1) has been selected as reference for the parametric studies in this thesis.



**Figure 5.1: Axisymmetric 3-D nozzle for I-Level project (by courtesy of [216])**

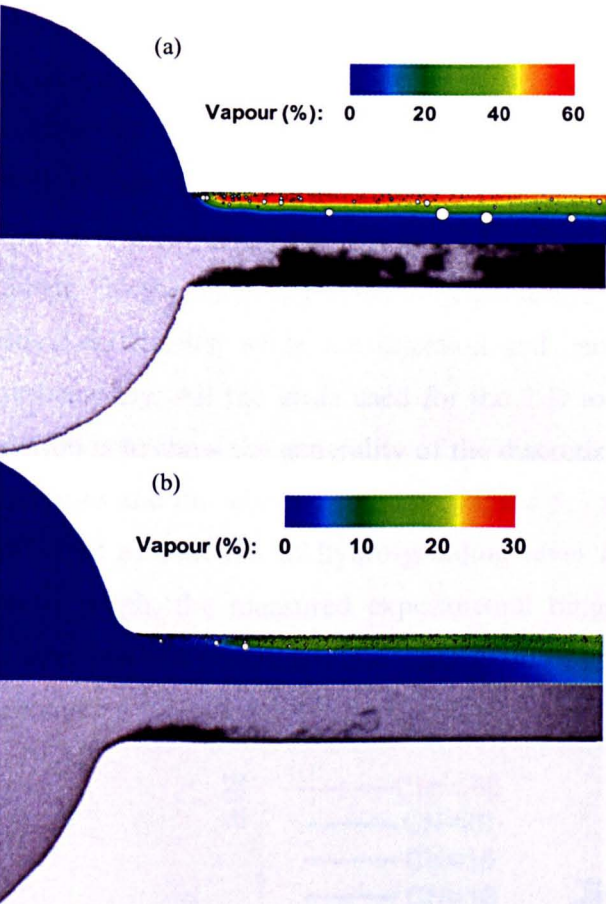
It is well known that the inlet edge rounding of nozzle holes has a dramatic effect on cavitation [18]; investigation of this effect was one of the key objectives of the I-Level project. For this axisymmetric geometry, three nozzles with varying injection hole inlet edge rounding were manufactured. The inlet rounding of each nozzle is represented by the degree of hydro-grinding or hydro-erosion (HE) of the nozzles; zero hydro-grinding (0%HE) represents a sharp inlet edge, whereas increasing degrees of hydro-grinding result in increased rounding of the hole inlet. Three different levels of hydro-grinding were selected, namely 0% HE, 9.5% HE and 18% HE (Figure 5.2).



**Figure 5.2: Real-size single-hole nozzles with varying degrees of hydro-grinding**

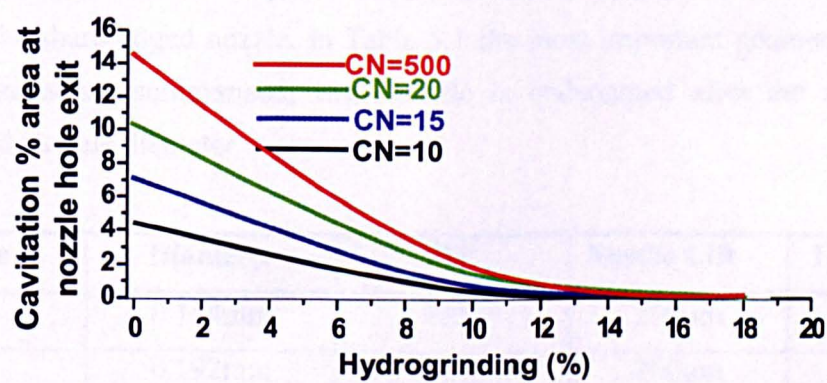
For this injector experimental and computational results have been presented recently [18, 180, 217]. Representative results of the cavitation appearing in the flow are shown in Figure 5.3. This geometry is relatively simple but of relevance to the physical processes investigated here.





**Figure 5.3: Single-hole nozzles for (a) 0% and (b) 9.5% hydro-grinding. The calculated void fraction distribution is shown for comparison with the CCD images, taken from [180].**

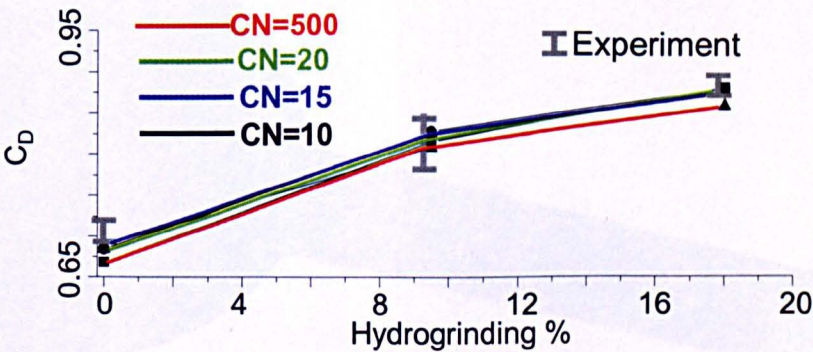
Clearly, changing the hole inlet radius of curvature, the cavitation intensity is modified. That also affects the nozzle discharge coefficient which varies considerably between the non-cavitating and the fully cavitating cases. Figure 5.4 shows the predicted blockage (dimensionless cross sectional area at the nozzle hole exit occupied by cavitation bubbles), as calculated by a recently developed cavitation model [217].



**Figure 5.4: Predicted nozzle hole exit blockage**



As it has been shown in a number of studies, for example [218], the cavitation number is the major dimensionless number used to characterise the onset of cavitation, the flow regimes formed inside the injection hole and the reduction of the discharge coefficient with increasing cavitation levels. The values tested in the I-Level project correspond to injection pressures from 300 to 1200bar, while the back pressure ranges from 20 to 54bar. For the particular cases to be presented here, the numerical grid was consisted of about 6,000cells, while the injection and back pressures used were 1000 and 20bar, respectively. All the grids used for the 2-D axisymmetric case were unstructured. The reason is to show the generality of the discretisation scheme that calculates the grid sensitivities and the adjoint gradients. Figure 5.5 shows the predicted nozzle discharge coefficient as function of hydro-grinding level for different cavitation numbers. On the same graph, the measured experimental range values are also shown. As clear, the model predicts reasonably the volumetric flow rate as function of the nozzle hole entry geometry.



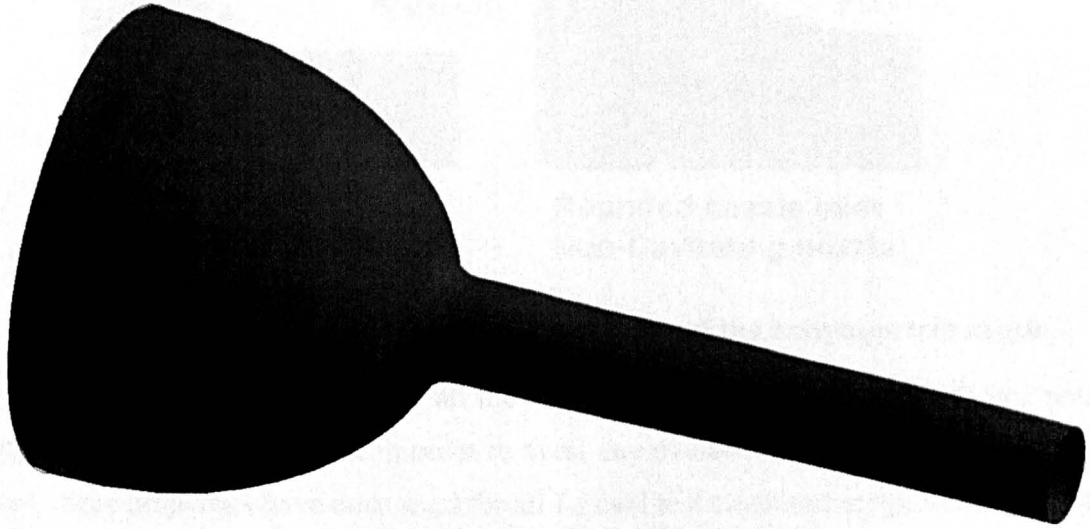
**Figure 5.5: Predicted discharge coefficient as function of inlet radius of curvature and cavitation number**

Due to this difference in the level of hydro-grinding, each nozzle hole has also a different diameter, in an effort to keep the flow rate of the hydro grinded nozzles comparable to that of the sharp-edged nozzle. In Table 5.1 the most important geometric details of all the nozzles are summarised; each nozzle is codenamed after the manufacturer (Bosch) and its hole diameter.

Nozzle	Diameter	HE	Needle Lift	Hole Length
B184	0.184mm	18%	200µm	~0.91mm
B192	0.192mm	9.5%	200µm	~0.96mm
B205	0.205mm	0%	200µm	1.01mm

**Table 5.1: Geometrical details of the I-Level nozzles**

For the parametric studies in this chapter we investigate these nozzles numerically in two-dimensions. Only half of the geometry is numerically simulated due to the symmetry of the real-size nozzles along their main axis. This decision was imposed by the fact that many runs were necessary for the parametric studies and the computational time increases significantly for 3-D simulations. The aforementioned I-Level nozzles, namely the sharp-edged (0% HE) axisymmetric nozzle at 200  $\mu\text{m}$  needle lift has been used as reference geometry for the parametric studies. Similar to the hydro-grinding geometries are reproduced using the parameterisation scheme developed for the purposes of 2-D nozzle design. An example of the sac and hole of the aforementioned geometry parameterised using the developed method (section 3.5.2) is presented in Figure 5.6.



**Figure 5.6: Example of parameterised I-Level nozzle**

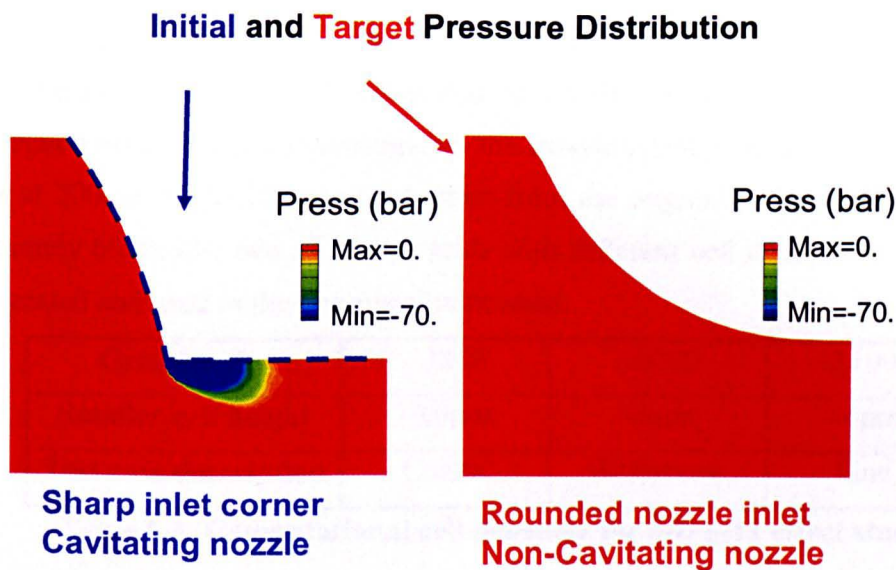
From the available experimental data [216] it was known that the reference nozzle is a highly cavitating one, and that for even very low cavitation numbers ( $CN \geq 9$ ) cavitation appears in the form of a film that extends to the exit of the hole.

The cost function (3.3) has been used throughout the parametric studies as this is the validation cost function of the developed method. The cost function expression is repeated here to facilitate the reader:

$$I_C = \frac{1}{2} \oint_w (p - \bar{p}_t)^2 ds - \oint_w (p - \bar{p}_t) \mu \frac{\partial \mathbf{u}}{\partial \bar{n}} \cdot \bar{\mathbf{n}} ds, \quad (3.3)$$



Another geometry has been constructed from the reference geometry using the parametrisation scheme described in section 3.5.2. The parameters used were 20% conicity and 25 $\mu$ m radius of curvature producing nozzle geometry similar to the 18% hydro-grinding one. The latter served as the means to calculate the target flow distribution using the **GFS** flow solver. In Figure 5.7 the initial and target pressure fields are shown.



**Figure 5.7: Initial and Target pressure fields of the axisymmetric nozzle**

The Diesel fuel that was used for all the experiments by all project partners had predetermined constant properties, in order to avert any deviations due to varying fuel qualities; these properties have been used for all I-Level test cases and are presented in Table 5.2 below.

Liquid density $\rho_L$	Dynamic viscosity $\mu$	Surface tension $\sigma$
835 kg/m <sup>3</sup>	2.5x10 <sup>-3</sup> kg/m·s	~ 0.02 N/m

**Table 5.2: Properties of the I-Level project Diesel fuel**

Results the developed method for the 2-D case described, using only the radius of curvature as parameter, have been presented by the author et al in [219, 220].



## 5.3 EFFECT OF NUMERICAL PARAMETERS

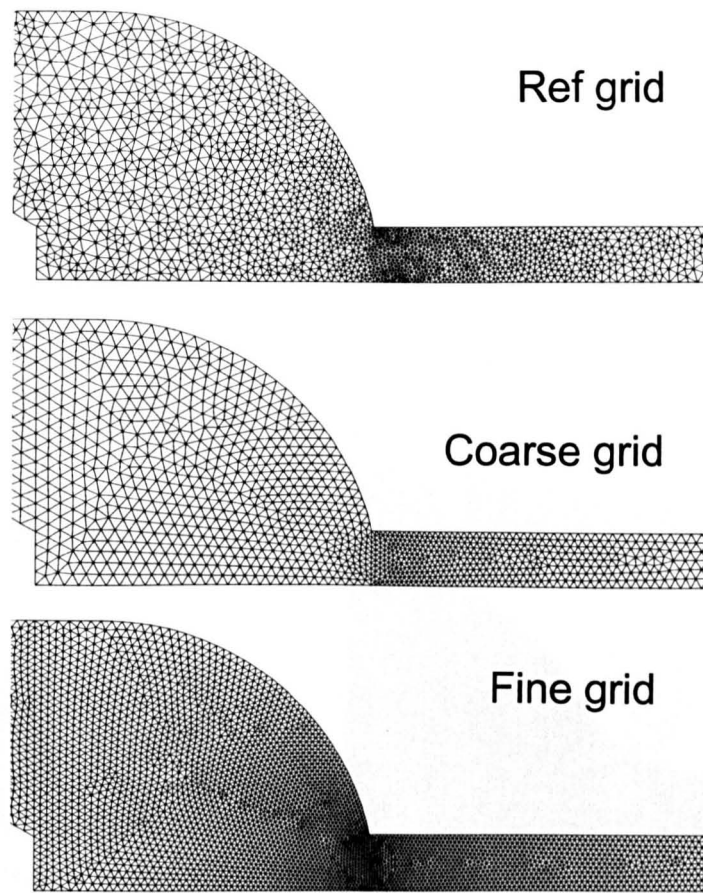
### 5.3.1 Effect of Grid

It is a well-known fact that the density in computational cells of the employed grid has a great impact on the accuracy of fluid flow simulations [204]. The use of a grid with very low cell density can cause failure to capture essential flow characteristics, especially in regions of the flow where large gradients are present. Obviously, this effect is also present in the flow and adjoint variables and as a result in the gradients and the overall optimisation process. For its investigation the axisymmetric real-size sharp-edged B205 nozzle at 200 $\mu\text{m}$  needle lift was used; apart from the original 2-D mesh, which has approximately 6000 cells, two additional grids with different cell densities (see Table 5.3) were created and used in the optimisation process.

<b>Grid density</b>	~3800	~6000	~11000
<b>Smaller cell height</b>	~10 $\mu\text{m}$	~6 $\mu\text{m}$	~4 $\mu\text{m}$
<b>Test case description</b>	Coarse	Reference	Fine

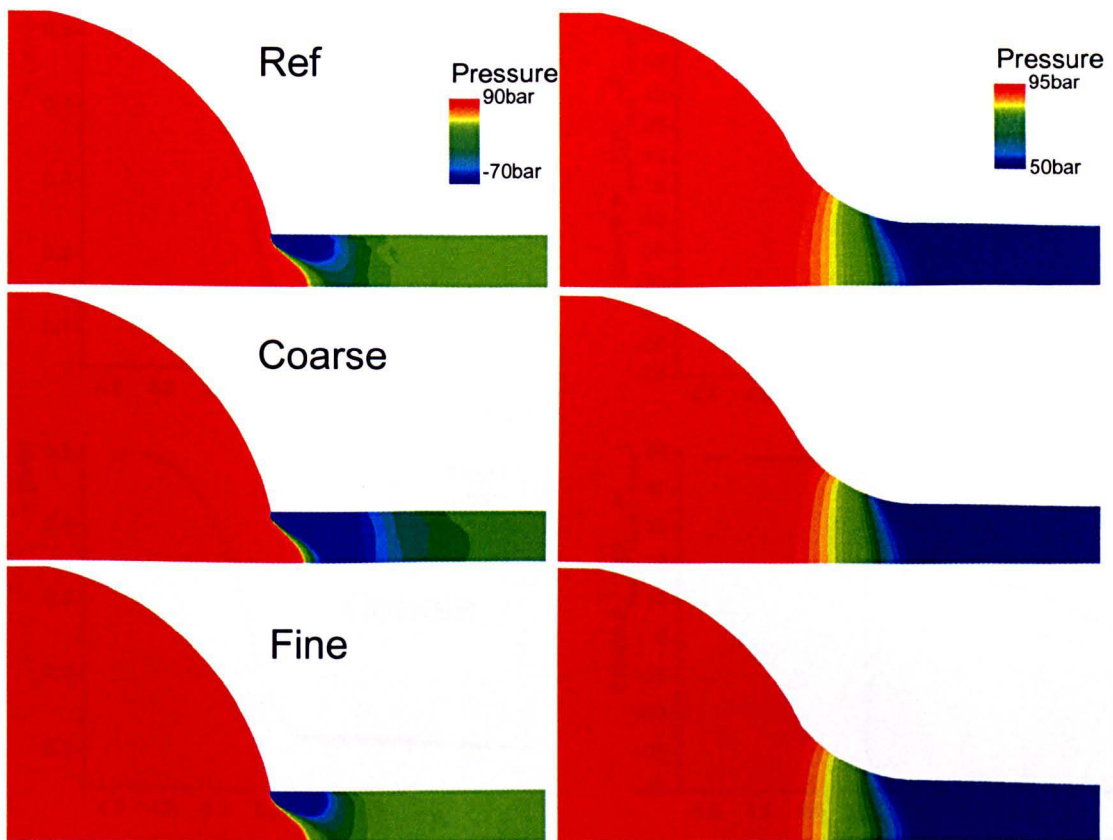
**Table 5.3: Computational cell densities for 2-D grid effect study**

The hole inlet is the location where the steepest pressure drop is expected to take place, and where cavitation is most likely to initiate as a result of this pressure drop. Thus, the created grids have a substantially increased cell density in the vicinity of the hole entrance, where the smaller size cells appear. The coarse grid is chosen in a way that the cell size is large but  $y^+$  is such that wall functions can be applied. The prescribed characteristics of the three grids are shown in Figure 5.8.



**Figure 5.8: Three grids used for the grid effect parametric study**

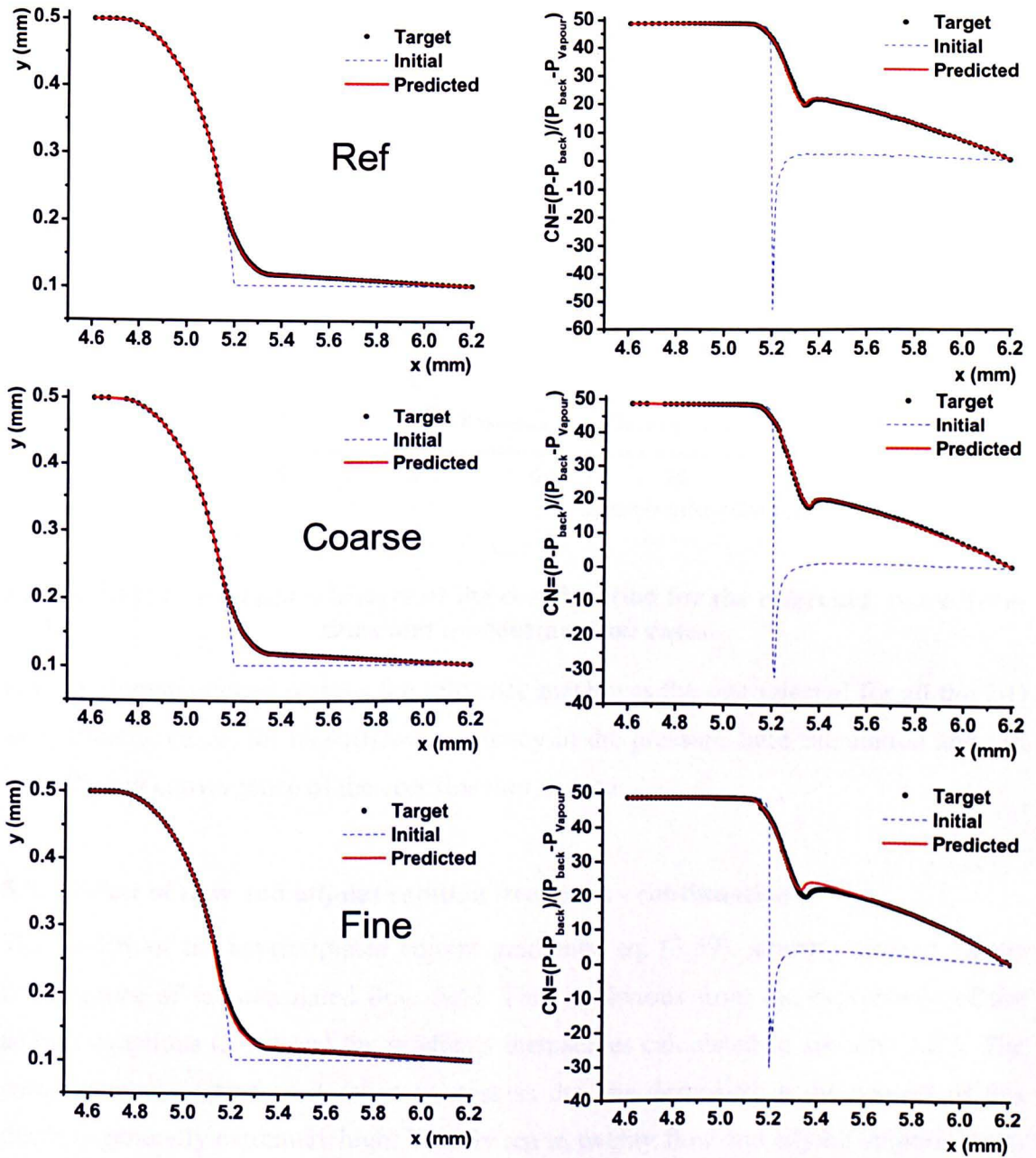
The target geometry, for all three cases, was also produced following the procedure described in section 5.2; by parameterising the initial geometry using a radius curvature and a conicity of almost 20%. The parameterised shapes are also slightly affected by the density change of the grid; so the target geometry is not identical for all discretisation cases. This is due to the fact that the parameterisation takes place after the discretisation of the geometry. The target pressure distributions, for the three different spatial discretisation cases, were calculated along with the initial one using the **GFS** flow solver. The calculated pressure fields are presented in Figure 5.9. Clearly the reference and fine grids seem to produce flow fields that are similar while the resulting pressure field of the coarse grid seem to be different field from the other two.



**Figure 5.9: Pressure fields for the initial and target geometries using the three different grids**

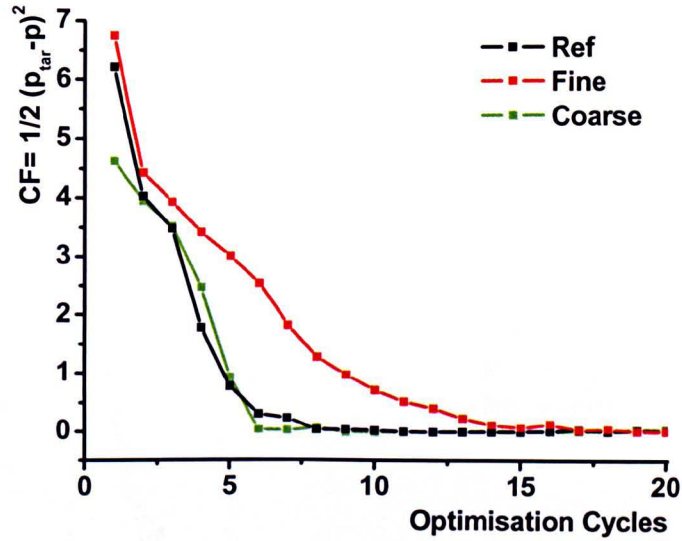
The same Figure 5.10 shows the corresponding pressure target distribution along the boundary of the nozzle; this part of the nozzle that is modified during optimisation. On the same graph, the pressure distribution corresponding to the initial nozzle design is also plotted. The developed optimisation model was then employed to re-construct the geometry of the targeted pressures. The corresponding results are also presented on the same Figure 5.10. As it is clear, after a few iterative solutions for the flow field and the adjoint equations, the code converges to the desired pressure and nozzle shape for all the different grids. The same figure shows also the initial geometry, the targeted geometry and the geometry predicted by the computer model after convergence for the three different grids investigated. As it is clear, the model predicts well the specified pressure distribution and nozzle geometry. In the case of the fine grid the pressure is not matched exactly. This is due to the fact that the more fine distribution of points along the boundary results to not smooth intermediate geometries during the optimisation process.





**Figure 5.10: Initial and targeted nozzle designs and dimensionless (CN) pressure vs. model predictions after convergence for the three grids**

Finally, Figure 5.11 shows the convergence history for the three test cases presented here. As can be seen, for all three grids, the model converges after a few iterations and stabilises to almost zero cost function value. As a conclusion all cases converge to their optimum solution and the method works for various spatial discretisations. The convergence rate in the case of fine mesh differs from the other two for the aforementioned reason.



**Figure 5.11: Convergence history of the cost function for the reference, more iterations and no-continuation cases**

For the aforementioned reasons the reference mesh was the one selected for all the 2-D axisymmetric cases; for its sufficient accuracy in the pressure field calculation and fast and efficient convergence of the cost function to zero.

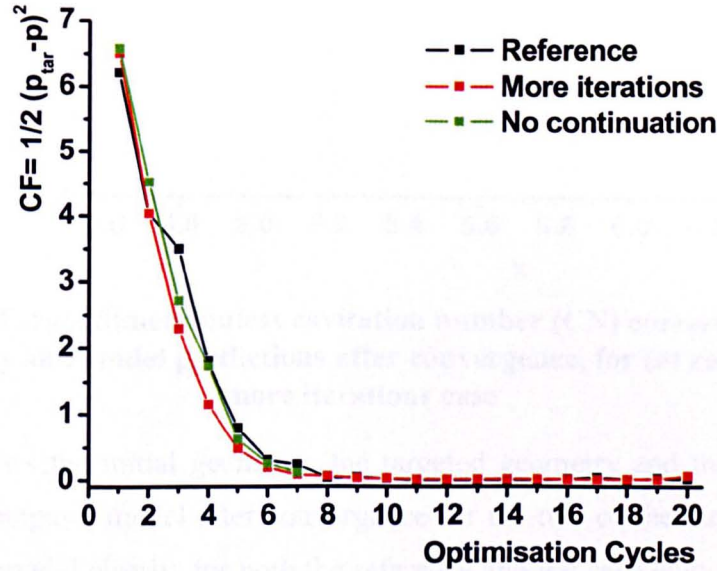
### 5.3.2 Effect of flow and adjoint solution iterations - continuation

The quality of the approximated adjoint gradients, eq. (3.59), strongly depend on the convergence of the calculated flow field. This is obvious from the expressions of the adjoint equations (3.48) and the gradients themselves calculated in sections 3.8.5. The computational cost of an iterative process as the one described in the context of this thesis is generally extremely high. Usually ten to twenty flow and adjoint solution iterations are necessary to achieve the optimum solution. Due to this fact continuation in the solution of the flow field has been implemented during the optimisation iterations. The continuation works in the sense that the flow solution of the modified geometry is not initialised but uses as initial values the flow field of the previous geometry. In this way the computational cost drops significantly; some times even 90%.

Two tests have been conducted, regarding the effect of the convergence of the flow and adjoint solution. For the first test the iterations of the flow and adjoint solution in each optimisation cycle have been increased from 300 to 1000. The overall computational time increased over 3 times. For the second test there was no continuation



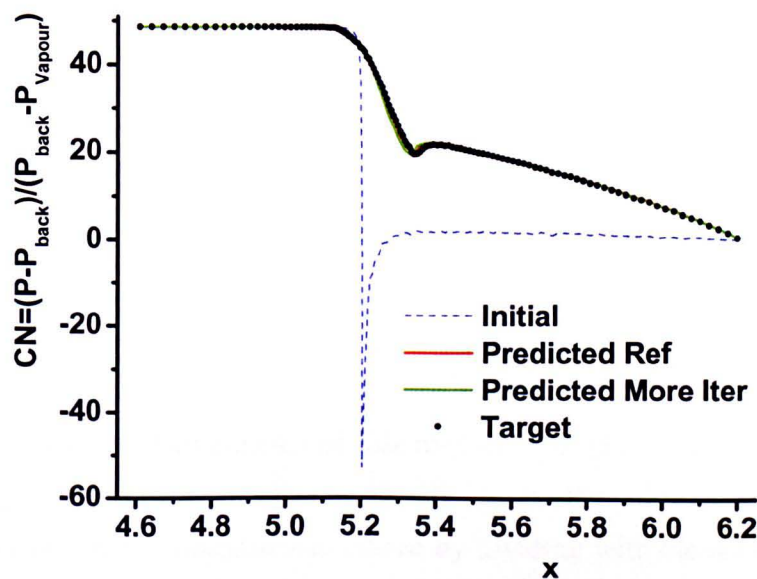
used in the flow solution. As a result ~3000 iteration were necessary for the flow field to converge increasing the computational time 10 times. The reference optimisation case has been used for both these tests and results showed smoother convergence of the cost function. Nevertheless, the cost function converges to the same value in almost the same number of optimisation cycles (Figure 5.12).



**Figure 5.12: Convergence history of the cost function for the reference, more iterations and no-continuation cases**

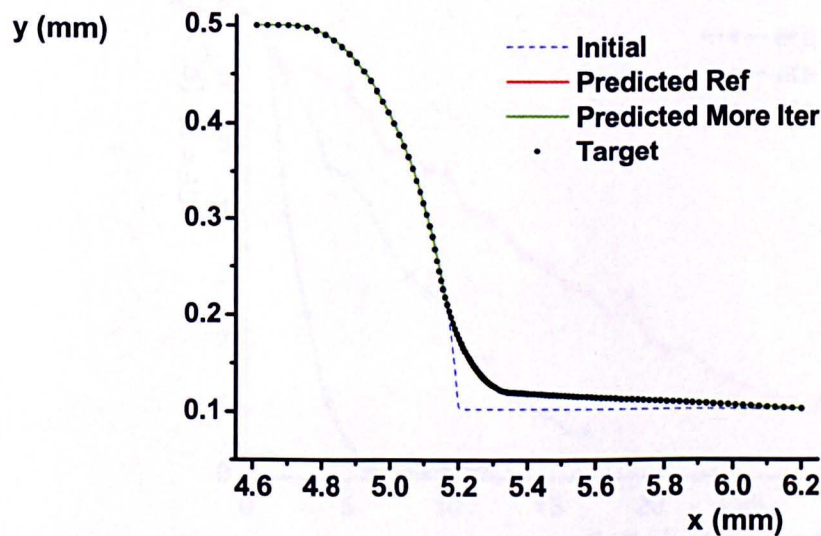
This is justified by the fact that after some optimisation cycles the actual convergence of the flow solution is not really different with continuation and without or if the flow iterations are increased. This is due to the fact that using continuation, as the iteration progress, the flow solution of each cycle starts from more converged values. Nevertheless, only mild levels of convergence in the adjoint equation are necessary to obtain highly accurate gradient information [123]. Apparently, unlike finite differences, deep convergence of the adjoint equation is helpful but not obligatory for the success of the overall optimisation process.

The target and initial pressure distributions of the reference and the case with more flow iterations are plotted in Figure 5.13. The model was employed to re-construct the geometry of the targeted pressure distribution. The predicted results are also presented on the same Figure 5.13 showing agreement with the expected/target pressure.



**Figure 5.13: Target dimensionless cavitation number (CN) corresponding to an initial geometry and model predictions after convergence, for (a) reference and (b) more iterations case**

Figure 5.14 shows the initial geometry, the targeted geometry and the geometry predicted by the computer model after convergence for the two of the three test cases investigated. The model clearly, for both the reference and the case with more flow iterations, predicts the specified pressure distribution and the desired nozzle geometry.



**Figure 5.14: Initial, target and predicted nozzle shapes for (a) reference and (b) more iterations case**

Due to this fact, continuation has been used in all cases and the flow and adjoint solutions were carried out with the least possible iterations. This led to significant computational time decrease with no actual loss in the quality of the predicted results.



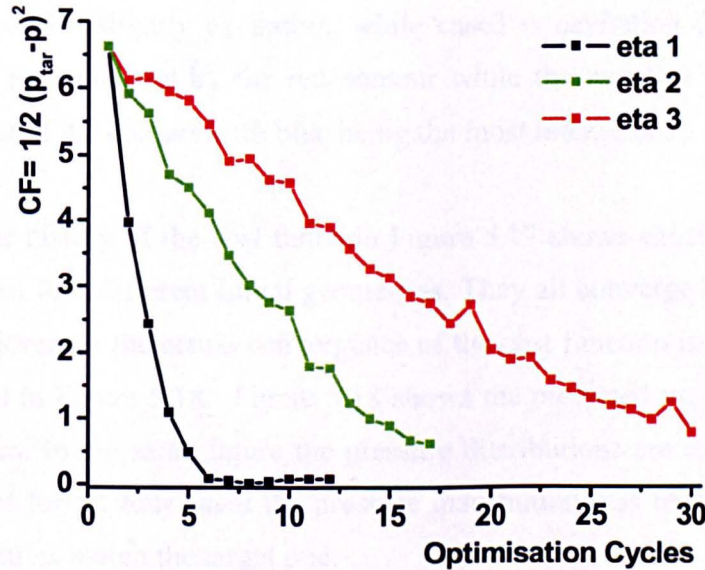
## 5.4 EFFECT OF OPTIMISATION PARAMETERS

### 5.4.1 Effect of Steepest descent step size

In section 3.11 the steepest descent method that was used in this study, for updating the design variables, was discussed. The expression of steepest descend scheme is repeated here for completeness, reading:

$$D_i^{new} = D_i - \eta \frac{\partial I_{aug}}{\partial D_i} \quad (4.87)$$

The efficiency and fast convergence of this method strongly depends in the choice of the step size  $\eta$  values. There are various ways to choose the value of the step size  $\eta$ . As we already discussed,  $\eta$  is un-dimensionalised by dividing with the normalised gradient (3.67). The more suitable value for the step size has been investigated in this section. Three choices of  $\eta$  were made for the reference test case; one was the same order of magnitude as the design variables (eta2) and the other two cases were 50% more (eta1) and 50% less (eta3). Results showed that the cost function of the case eta1 with the large step size converged faster and deeper than the other two choices (Figure 5.15). Even larger values of the step size were tested but failed to converge.



**Figure 5.15: Convergence history of reference geometry with effect of step size**

Throughout this study the step size eta1 was used for the 2-D test cases since it is proven to be the most suitable value for the convergence of the optimiser. For the 3-D cases step sizes of similar magnitude were used.



5.4.2 Effect of different initial geometries

Four different initial geometries were tested trying to reach the same target distribution. The characteristics of these geometries are summarised in Table 5.4. This test was conducted to assess the ability of the method to converge to the optimum solution regardless of the choice of initial geometry, thus design parameters.

	Case1 Initial	Case2 Initial	Case3 Initial	Case4 Initial (Ref)	Target
Conicity	10%	5%	0%	0%	20%
Radius of curvature	60µm	50µm	30µm	10µm	250µm

Table 5.4: Cases showing the effect of different initial geometries

The starting point of the optimisation procedure is rather critical in gradient based optimisation [35]. Moreover, from the results presented here it is clear that the developed method can perform excellent in all four cases. The reference case used in the parametric studies is case4 which as it can be seen in Figure 5.16 is the sharp geometry with the most intense cavitation inception. In the same Figure 5.16, the pressure field and distribution along the boundary for all four cases is presented as well as the geometry of the parameterised 2-D nozzle. Apart from the reference case which is highly cavitating cases1 and 2 are also slightly cavitating, while case3 is cavitation free. The positive pressure values are indicated by the red contour while the negative values are represented by the rest of the colours with blue being the most intense ones.

The convergence history of the cost function Figure 5.17 shows excellent behaviour of the method for all four different initial geometries. They all converge before 10 optimisation cycles. Moreover the actual convergence of the cost function is obvious from the results presented in Figure 5.18. Figure 5.18 shows the predicted vs. target geometries for the four cases. In the same figure the pressure distributions are superimposed. The results show that for all four cases the pressure distribution was reproduced while the predicted geometries match the target one.

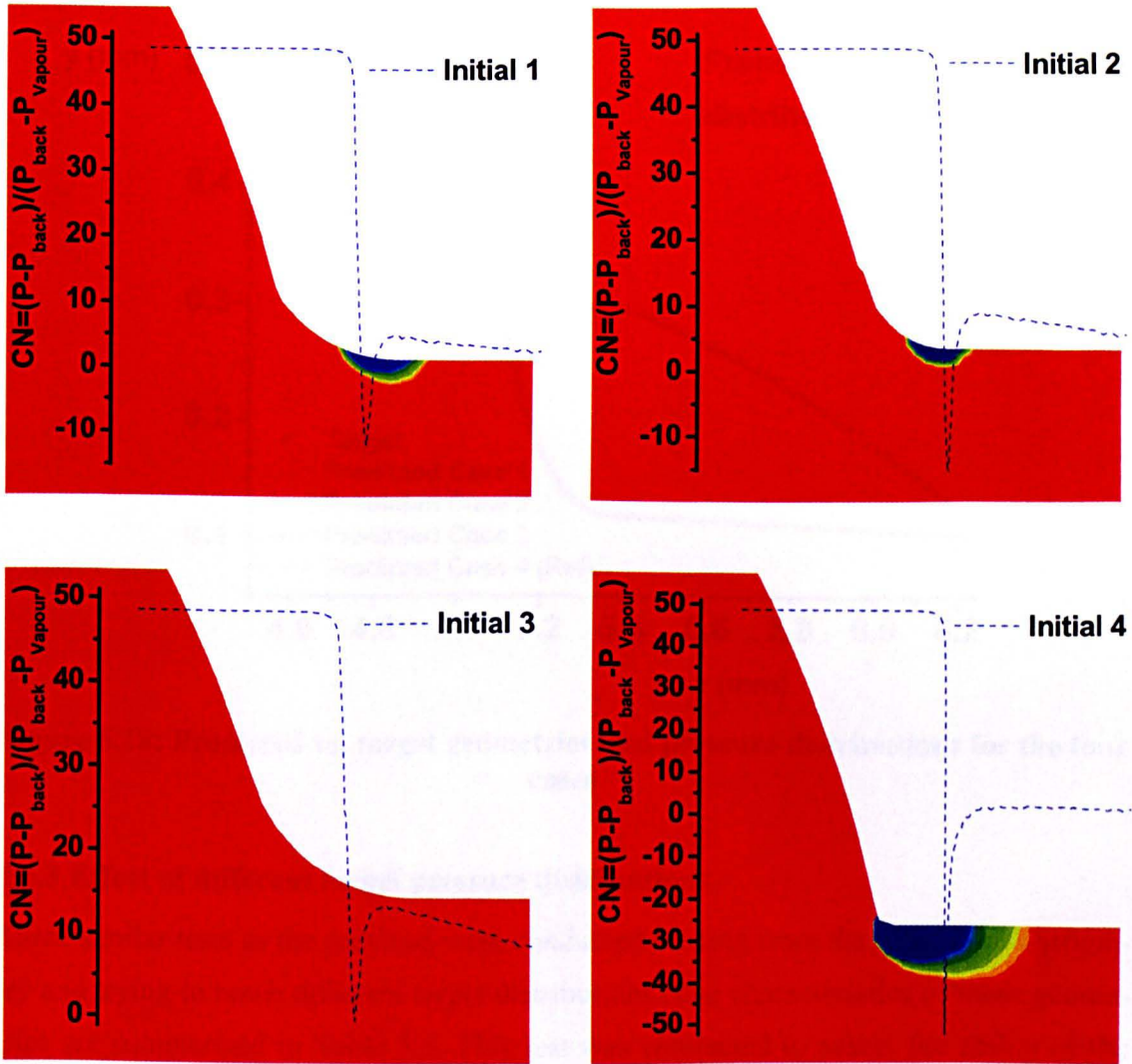


Figure 5.16: Geometry, pressure field and distribution for the four different initials

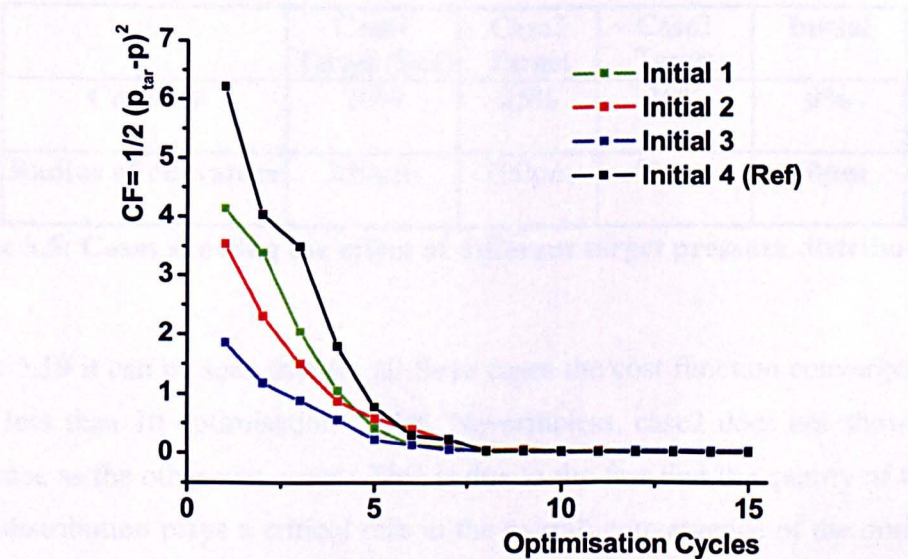


Figure 5.17: Convergence history of validation case using different initial geometries

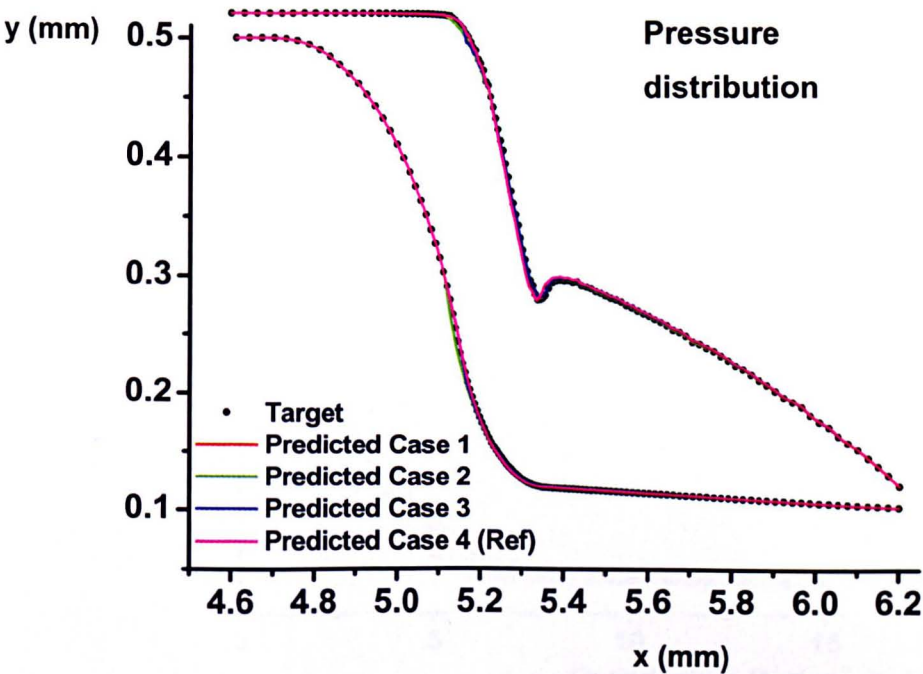


Figure 5.18: Predicted vs. target geometries and pressure distributions for the four cases

5.4.3 Effect of different target pressure distributions

Three similar tests as the previous were conducted starting from the same initial geometry and trying to reach different target distributions. The characteristics of these geometries are summarised in Table 5.5. This test was conducted to assess the ability of the method to converge to the optimum solution regardless of the choice of target.

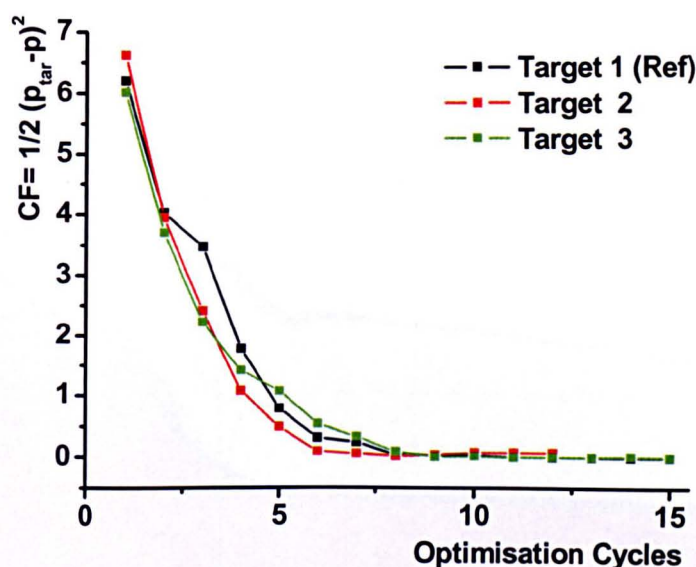
	Case1 Target (Ref)	Case2 Target	Case3 Target	Initial
Conicity	20%	25%	10%	0%
Radius of curvature	250μm	200μm	400μm	10μm

Table 5.5: Cases showing the effect of different target pressure distributions

In Figure 5.19 it can be seen that for all three cases the cost function converged to zero value in less than 10 optimisation cycles. Nevertheless, case2 does not show as deep convergence as the other two cases. This is due to the fact that the quality of the target pressure distribution plays a critical role in the overall convergence of the optimisation method. From Figure 5.20 it is obvious that the geometry of the second case has a sharp corner on the sac volume caused by the part of cycle put there by the parameterisation

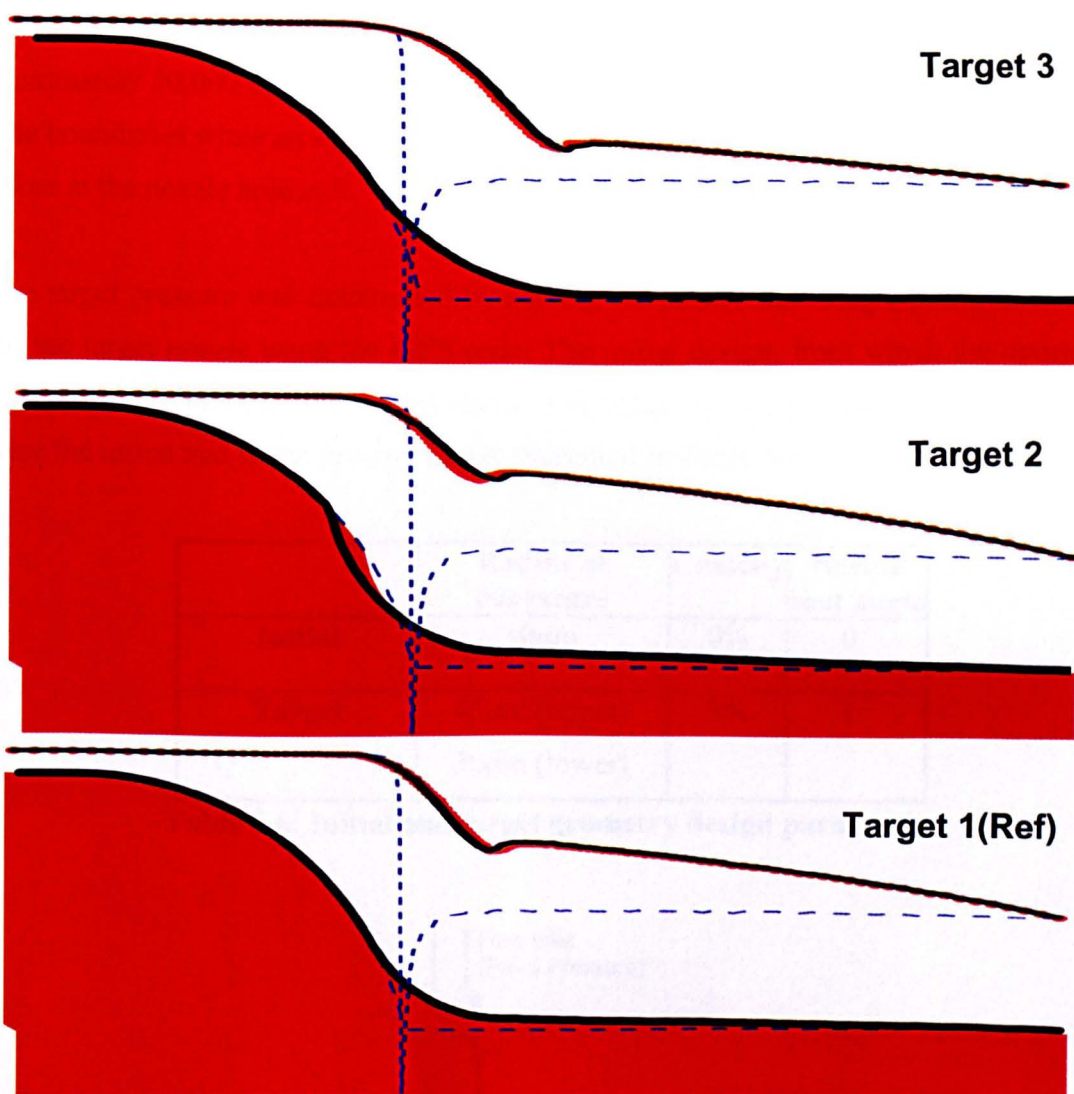


scheme. Something like that can happen for large values of the radius of curvature since in the parameterisation scheme we try to keep the sac volume shape unaffected.



**Figure 5.19: Convergence history of validation case using different target pressure distributions**

In the same Figure 5.20 the three target geometries are presented and the predicted (black) and initial (blue) geometries are superimposed to them. Also, in Figure 5.20 the initial, predicted and target distributions are shown using the same colours as for the geometries. Results show that in all cases the target pressure distributions were reproduced and the predicted geometries approximated the target ones; apart from already mentioned problem in case2. In the current study the first target distribution has been used as reference for the parametric studies. In the next section a more challenging 3-D validation case will be presented.



**Figure 5.20: Geometry, pressure field and distribution predicted (black) vs. target (red) for the three different target cases. Initial (blue) geometry and pressure distribution**

### 5.5 INVERSE DESIGN OF 6-HOLE VCO NOZZLE

The scope of the present study is the optimisation of real Diesel nozzles. For this reason, more realistic, 3-D cases are necessary for the validation of the method. The 3-D test validation case consists of matching a pre-described pressure distribution at the hole inlet of a typical VCO multi-hole Diesel nozzle. This test case has been recently presented by the author et al in [221] but in that case only the radius of curvature was used as design parameter. The hole diameter of the nozzle is  $152\mu\text{m}$  and the hole length  $970\mu\text{m}$ . The geometry simulated corresponds to only one of the assumed six holes. Furthermore, only half of it has been simulated, thus the computational domain corresponds



to 1/12<sup>th</sup> of the whole nozzle design, as shown in Figure 5.21 and it consists of approximately 50,000 cells. Symmetry boundary conditions have been used at the left and side boundaries while an injection pressure of 1000bar has been assumed at the inlet and 20bar at the nozzle hole exit.

The target pressure was determined by solving the steady-state single-phase flow field for the target nozzle using the **GFS** code. The initial design, from which the optimisation procedure started, was almost sharp. The values of the parameters used to reproduce the initial and target geometries are presented in Table 5.6.

	Radius of curvature	Conicity	Needle seat angle
Initial	10µm	0%	0°
Target	45µm (upper) 20µm (lower)	5%	1°

Table 5.6: Initial and target geometry design parameters

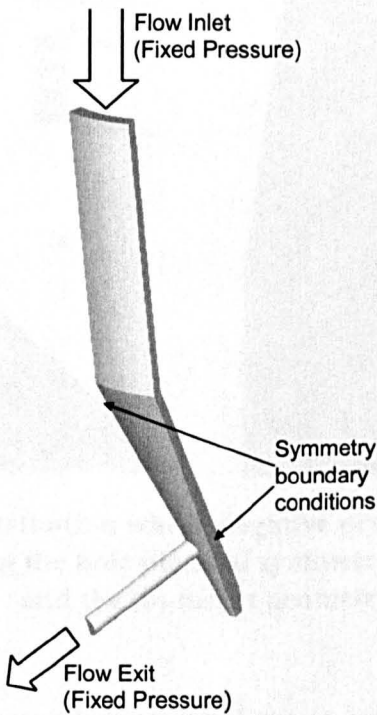
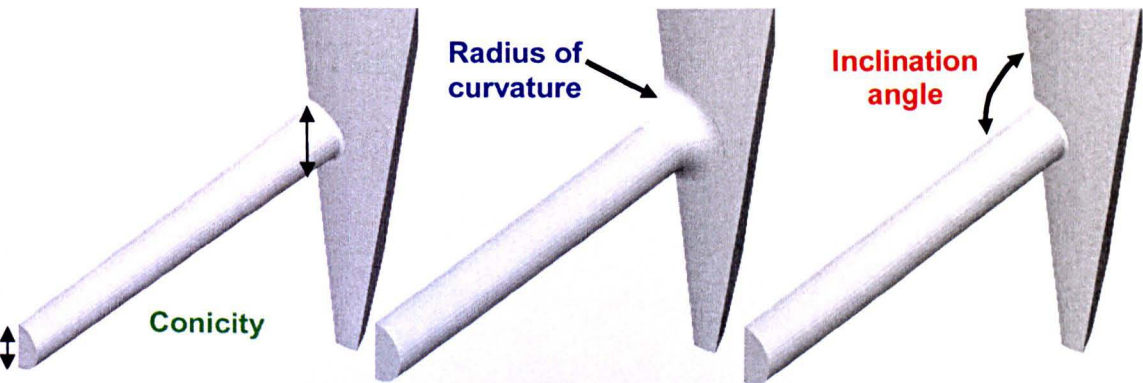


Figure 5.21: 3-D VCO 6-hole Nozzle geometry and test case simulated

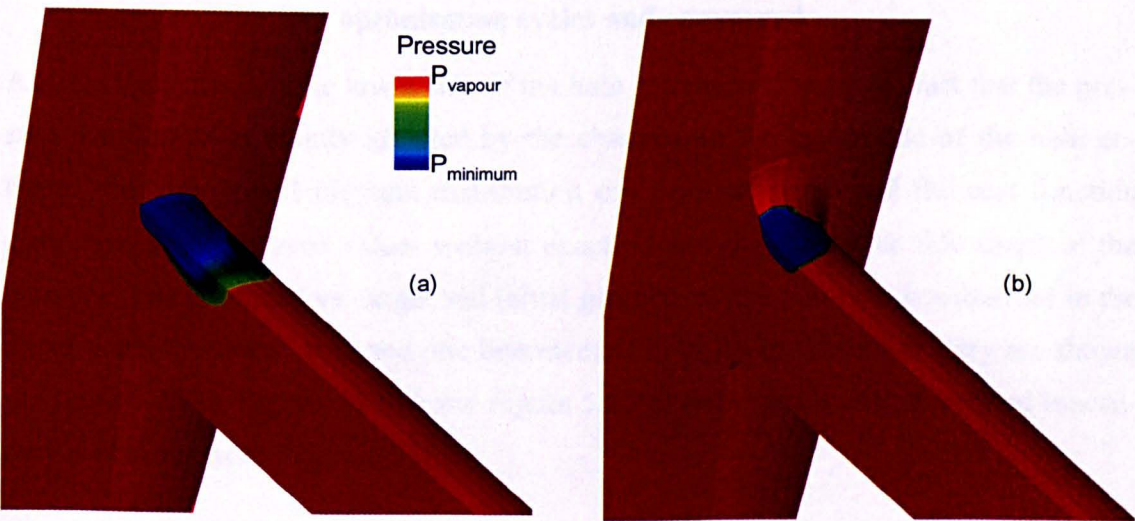
For the parameterisation of the aforementioned geometries the developed scheme (section 3.5.2) was used as it is depicted in Figure 3.5. Figure 5.22 shows how the parame-

terisation scheme was employed in the 3-D cases and the actual parts of the nozzle that are modified when each design parameter changes value.



**Figure 5.22: The three design parameters used for the reproduction of the 3-D geometries**

The pressure distribution for the initial and the target nozzle are represented in Figure 5.23 (a) and Figure 5.23 (b). The figure also shows the area of negative pressure which is reduced in the target geometry.

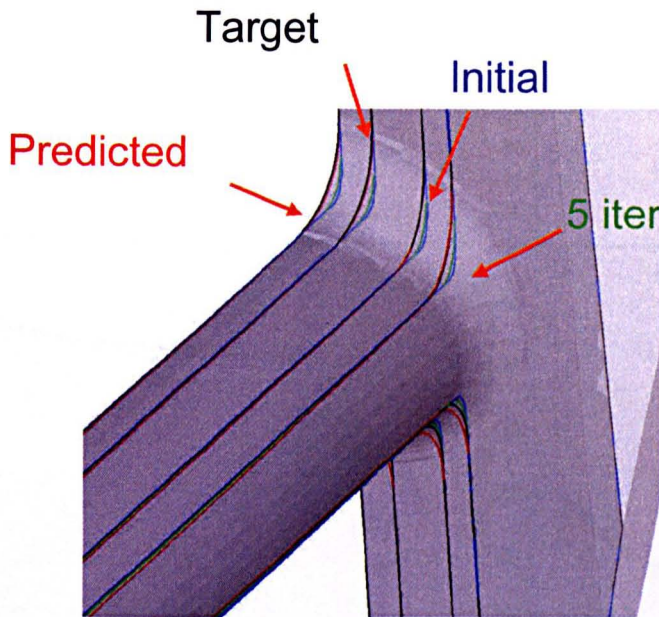


**Figure 5.23: Pressure distribution where negative pressure is depicted at the hole entry and flow field along the hole plane of symmetry of the (a) initial geometry and the (b) target geometry**

Figure 5.24 presents various nozzle boundary lines on four cross sections parallel to the hole plane of symmetry. Those correspond to the initial one, having the sharpest corners, the target one, which is coinciding with the predicted one after ~20 optimisation cycles as well as an intermediate one as predicted after 5 optimisation cycles. From the



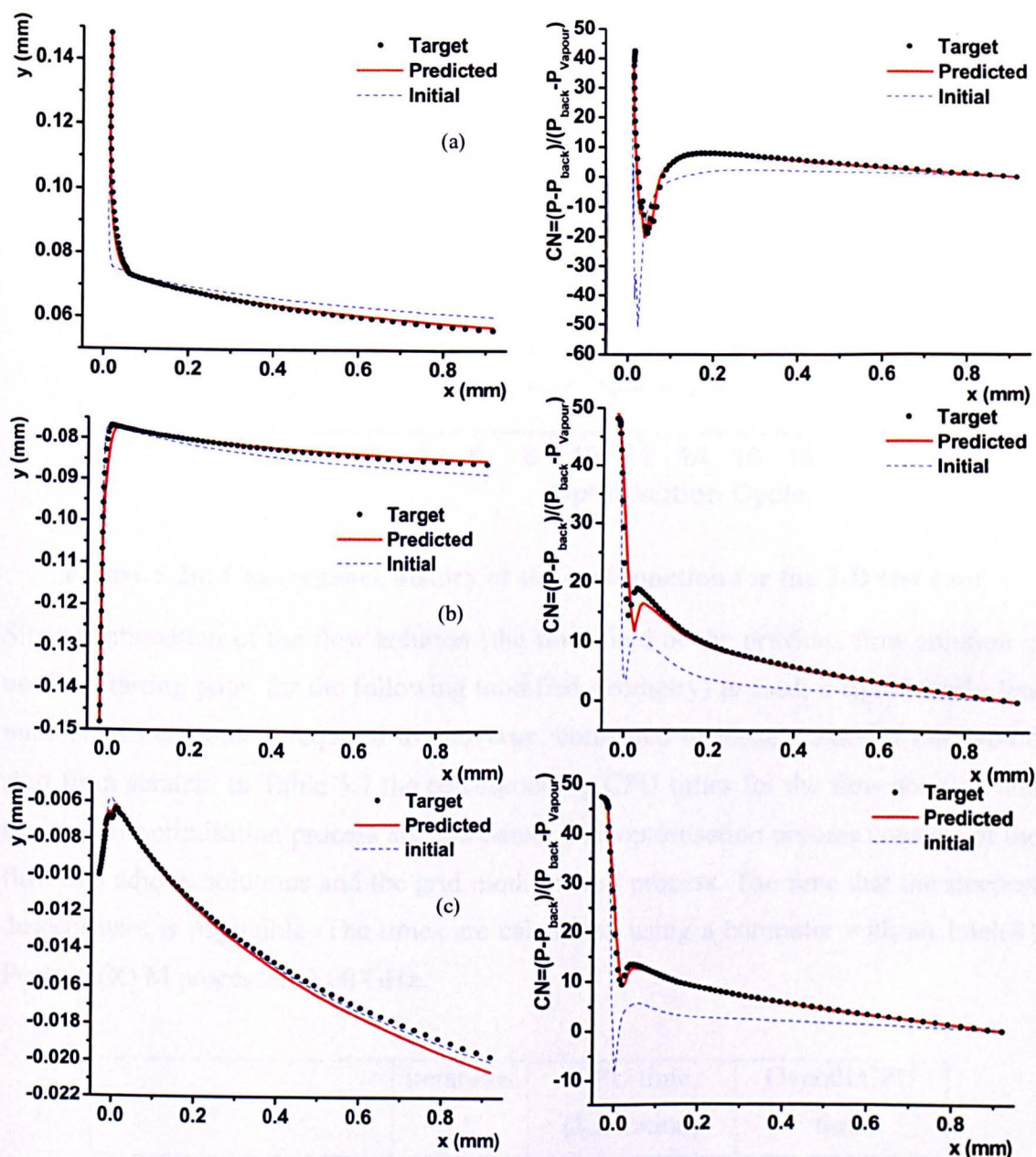
results shown in Figure 5.24, it is obvious that the model is able to predict the target geometry in the three dimensional space.



**Figure 5.24: Cross sections at four locations parallel to the hole plane of symmetry, showing the initial geometry, the target geometry, the predicted geometry after 5 optimisation cycles and converged**

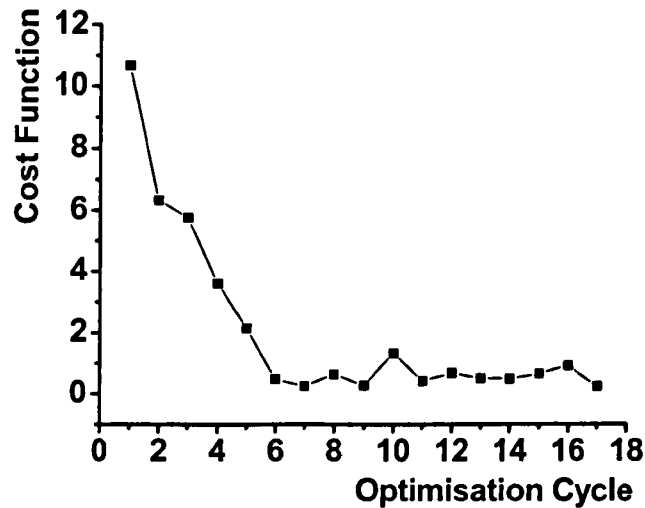
A slight inaccuracy in the lower side of the hole entrance is due to the fact that the pressure distribution is mainly affected by the changes on the upper side of the hole entrance. So, the overall pressure distribution can be reproduced and the cost function converges to nearly zero values without exactly matching the lower side shape of the entrance. The predicted vs. target and initial geometries and pressure distributions in the upper zone, the lower zone and one intermediate zone on the hole boundary are shown in Figure 5.25(a), Figure 5.25(b) and Figure 5.25(c) respectively. The described inaccuracy is obvious from Figure 5.25(b).





**Figure 5.25: Predicted vs. target and initial geometries and pressure distributions for the 3-D validation case in three 'zones' (a) upper, (b) lower and (c) middle**

This time, the number of cycles required for the cost function to converge is slightly bigger than in the previous 2-D cases. Still, as can be seen in Figure 5.26, the method converges after less than 20 cycles.



**Figure 5.26: Convergence history of the cost function for the 3-D test case**

Since continuation of the flow solution (the flow field of the previous flow solution is used as starting point for the following modified geometry) is used, a significantly less number of iterations is required to converge, compared to those needed if one would start from scratch. In Table 5.7 the corresponding CPU times for the flow solution and the overall optimisation process are compared. The optimisation process consists of the flow and adjoint solutions and the grid modification process. The time that the steepest descent uses is negligible. The times are calculated using a computer with an Intel(R) Pentium(R) M processor, 2.00 GHz.

	iterations	CPU time (1 iteration)	Overall CPU time
Flow solution (no continuation )	~5000	~6sec	~500min
Flow solution (continuation)	~300	~6sec	~30min
Adjoint solution	~300	~2sec	~12min
Optimisation	~20	~45min	~900min

**Table 5.7: Comparison of CPU time required for adjoint optimisation**

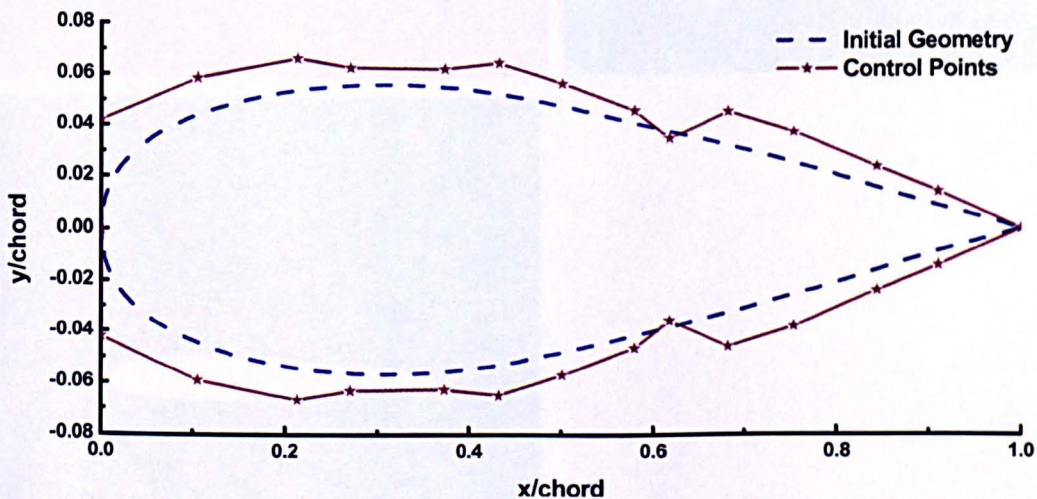
In this case the total computational time for predicting the target geometry corresponds to less than two direct flow field solutions.



## 5.6 INVERSE DESIGN OF AN ISOLATED AIRFOIL

In this section the inverse design of an airfoil is presented as a means to show the generality of the proposed optimisation method. For this test case the target pressure distribution  $C_p$  is obtained from a flow analysis of a best-fit of the *NACA0012* airfoil with angle incidence of  $0^\circ$  degrees and far field velocity  $u = 35\text{m/sec}$  to ensure subsonic flow because the flow solver only deals with incompressible flows.

Different arbitrary and non-dimensional starting geometries were tested as a parametric study to ensure the independence of the optimisation code to the initial geometry. For the parameterisation two Bezier curves were used: one for the pressure and one for the suction side. Each side of the airfoil is modelled separately by using Bezier polynomials represented by 15 control points as in Figure 5.27. The leading and trailing edge control points are kept fixed during the optimisation procedure to avoid rigid body movement. The second control point is also in  $x = 0$  so that the steep slope of the leading edge of the airfoil can be represented.



**Figure 5.27: Parameterisation of initial geometry using Bezier polynomials**

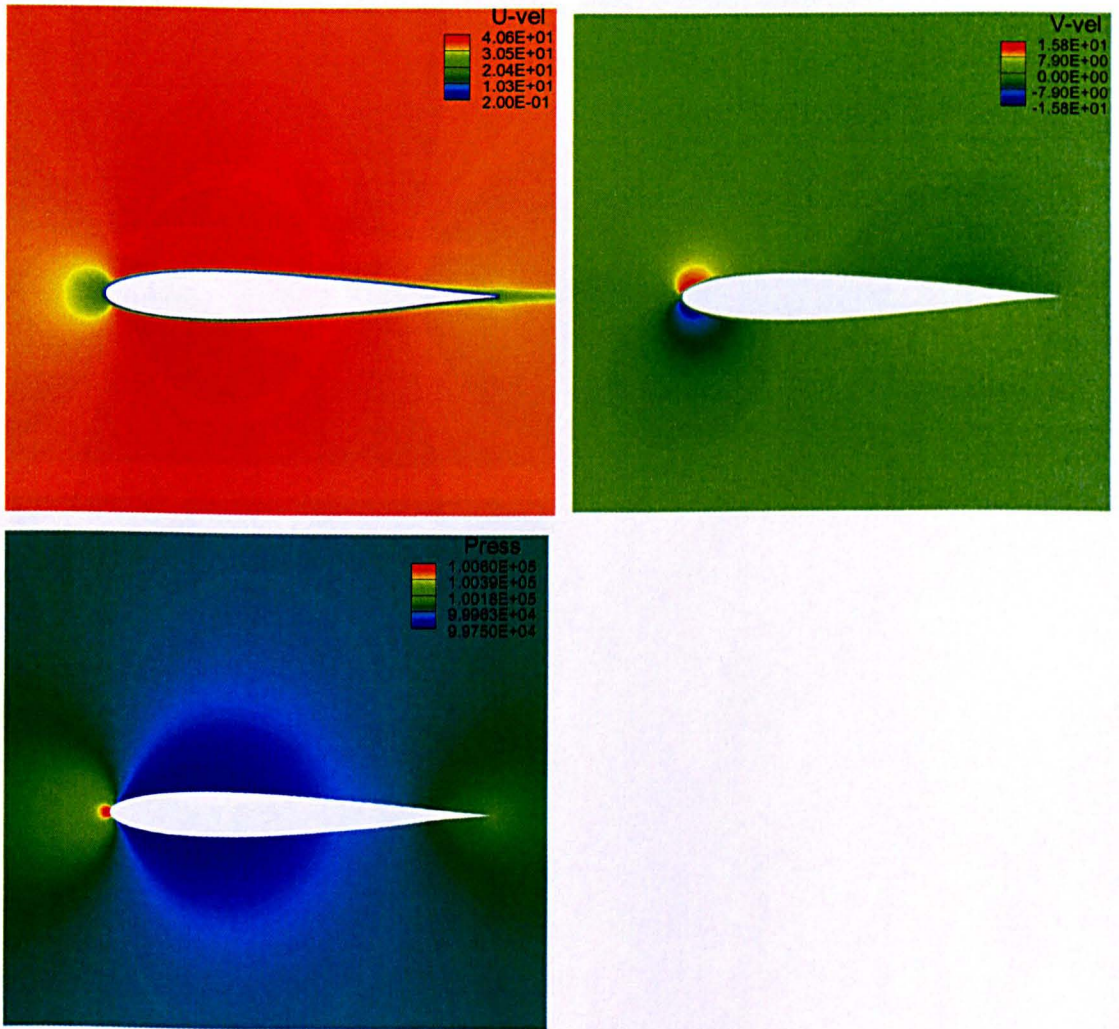
The required computational unstructured meshes are generated using a typical advancing front method. The grid generation software is very fast for the 2-D cases, so it is repetitively used within each cycle to create the computational grid for every new airfoil.

Two cases having different initial geometries are presented here. Both airfoils are symmetric and the cost function is represented by equation (3.3). As we have already



pointed out the second integral is equal to zero but is necessary for the extraction of the boundary conditions of the adjoint equations.

The convergence of the adjoint equations and, as a result, of the whole optimisation procedure, strongly depends on the convergence and accuracy of the flow solution. In Figure 5.28 we can see the  $u$  and  $v$  velocities field and the pressure field around the airfoil.



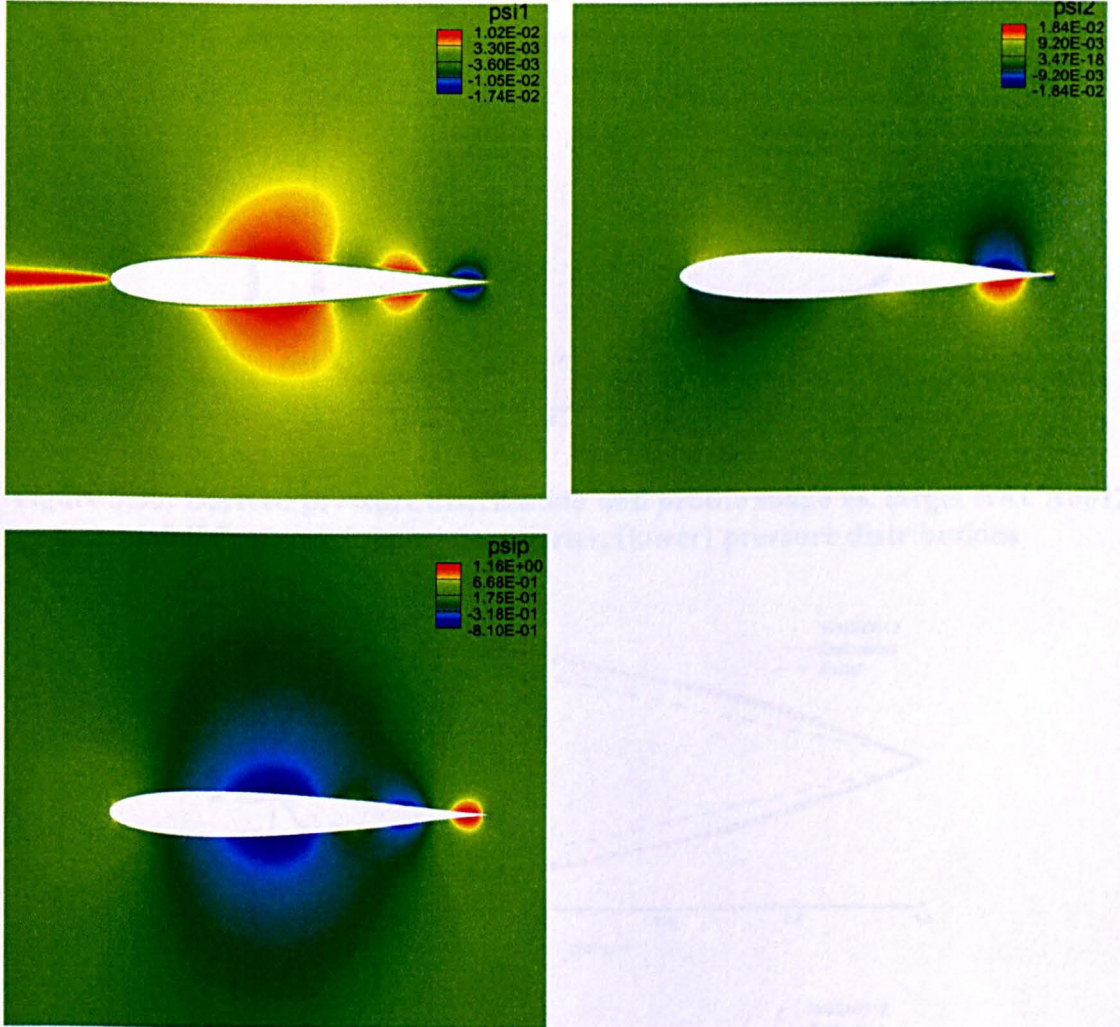
**Figure 5.28: Flow field around *NACA0012* airfoil (upper left)  $u$ -velocity, (upper right)  $v$ -velocity, (lower) pressure distributions**

In Figure 5.29 the corresponding costate variable fields are represented. The adjoint/costate variable fields have no physical meaning.  $\Psi_1$  and  $\Psi_2$  show the tension of



the geometry to be modified in the  $x$  and  $y$  direction respectively. As for the  $\Psi_p$  takes no part in the gradient but is the variable that works as the pressure in the pressure correction scheme used for the solution of the adjoint equations.

Comparison of the target pressure distribution and the *NACA0012* airfoil geometry with the predicted pressure distribution and airfoil geometry using the optimisation method can be seen in Figure 5.30 and Figure 5.31 for the two cases respectively.

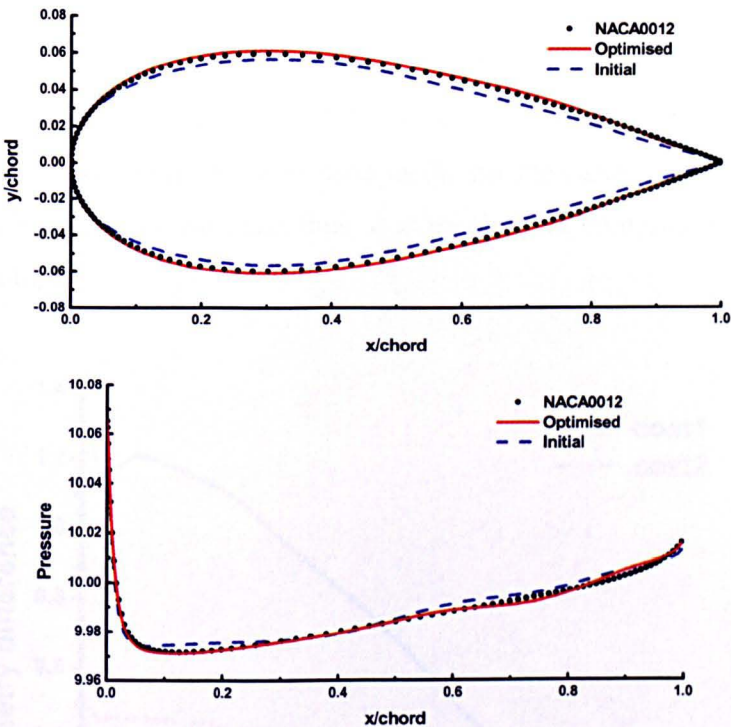


**Figure 5.29: Adjoint variables around NACA0012 airfoil (upper left)  $\Psi_1$ , (upper right)  $\Psi_2$ , (lower)  $\Psi_p$**

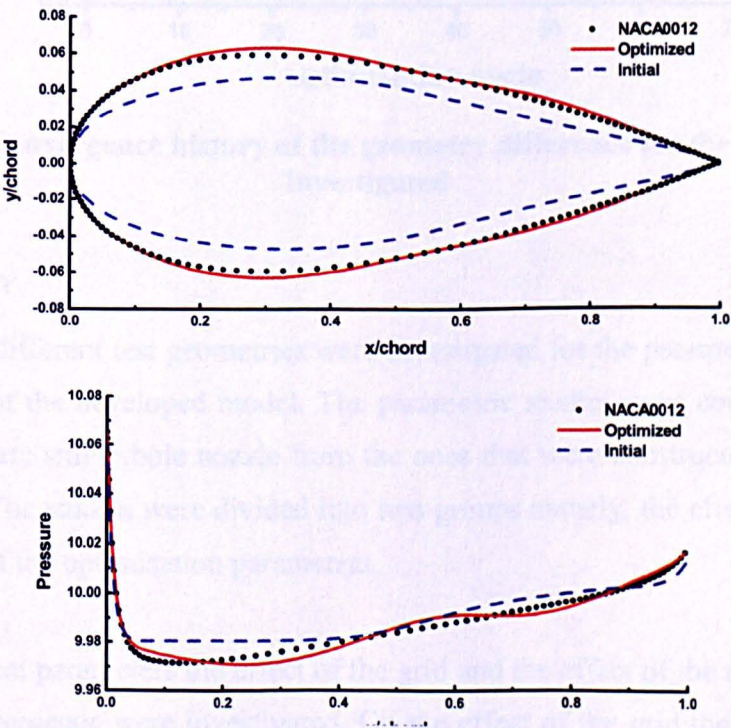
The purpose of selecting this test case is to investigate the accuracy level of the computed gradient, because the optimal solution (i.e. the target airfoil) is known beforehand. The computed gradient is considered to be of sufficient accuracy if the known/target optimal solution is obtained. In Figure 5.32 the convergence history of the cost function



shows that the prior goal is fulfilled and the value of the cost function has significantly dropped towards zero.



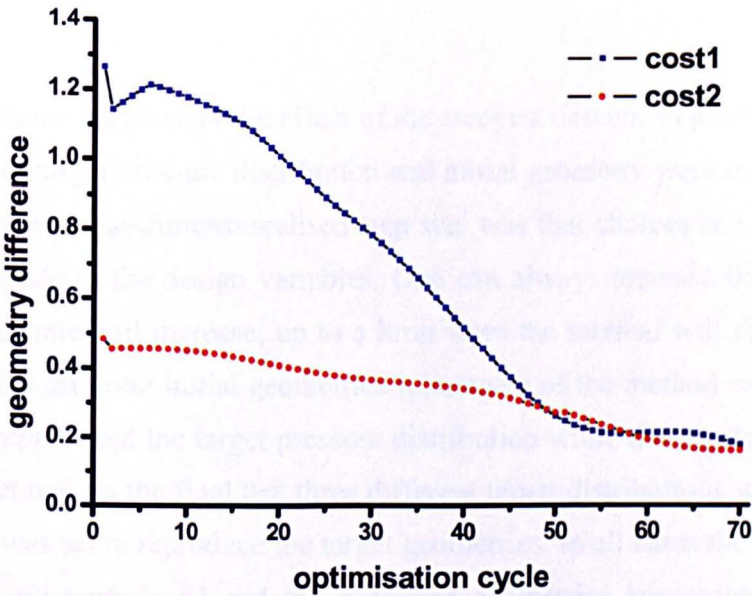
**Figure 5.30: Derived pressure distribution and profile shape vs. target NACA0012 airfoil first case (upper) geometries, (lower) pressure distributions**



**Figure 5.31: Derived pressure distribution and profile shape vs. target NACA0012 airfoil second case (upper) geometries, (lower) pressure distributions**



The presented results have been obtained in approximately in 70 optimisation cycles but it can be noticed that the most significant reduction of the cost has occurred during the first 50 cycles and after the convergence is stabilized. The optimisation cycles that are needed for this test case are significantly more than for the previous examined 2-D and even 3-D nozzle cases. The reason is that the design parameters that describe the airfoil geometry are much more than the ones used in the nozzle cases. Moreover, the dimension of the search space increases and thus, it is much more complicated to converge to an optimum solution.



**Figure 5.32: Convergence history of the geometry difference for the two test cases investigated**

### 5.7 SUMMARY

In this chapter different test geometries were investigated for the parametric studies and the validation of the developed model. The parametric studies were conducted using a 2-D axisymmetric single-hole nozzle from the ones that were constructed during the I-Level project. The studies were divided into two groups namely, the effect of numerical and the effect of the optimisation parameters.

For the numerical parameters the effect of the grid and the effect of the number of iterations until convergence were investigated. On the effect of the grid the conclusion can be drawn that all cases converge to their optimum solution and the method works for various spatial discretisations. The convergence rate in the case of fine mesh differs

from the other two cases due to the fact that the more fine distribution of points along the boundary results in not smooth intermediate geometries during the optimisation process. Moreover, the accuracy of the predicted pressure distribution of the coarse grid is poor. So, the reference grid is selected for the 2-D axisymmetric case. For the parametric study of the depth of convergence in the flow and adjoint solution all cases again matched the predescribed target pressure distribution. The case that used more iterations for the flow and adjoint solution had a smother convergence history. Nevertheless, the continuation scheme presented in the thesis was used for all the cases because it leads to significant computational time decrease with no actual loss in the quality of the predicted results.

For the optimisation parameters the effect of the steepest descent step size and the effect of the choices of target pressure distribution and initial geometry were investigated. The conclusion drawn for un-dimensionalised step size was that choices can be made in the order of magnitude of the design variables. One can always increase the step size and the convergence rate will increase, up to a limit where the method will fail to converge. In the case of the different initial geometries robustness of the method was shown since all four cases reproduced the target pressure distribution while the predicted geometries match the target one. In the final test three different target distributions were considered and the model was set to reproduce the target geometries. In all cases the target pressure distributions were reproduced and the predicted geometries approximated the target ones. Nevertheless, in case 2 the results were not as good as in the other cases because of the bad quality of the chosen target geometry. This led to the conclusion that the quality of the target plays an important part in the convergence of the method.

The 3-D test validation case consisted of matching a pre-described pressure distribution at the hole inlet of a typical VCO multi-hole Diesel nozzle. Four design parameters were used in order to reproduce the in-between and resulting geometries of the design process. The optimisation method converges to the predescribed pressure distribution in less than 20 cycles while the predicted geometry approximates really well the target one.

Finally the inverse design of an *NACA 0012* airfoil was conducted using as design parameters Bezier polynomials. Two different initial geometries were used to match the



predescribed pressure distribution. The predicted results show the generality of the proposed methodology.

# *Chapter 6*

## *METHOD APPLICATIONS*

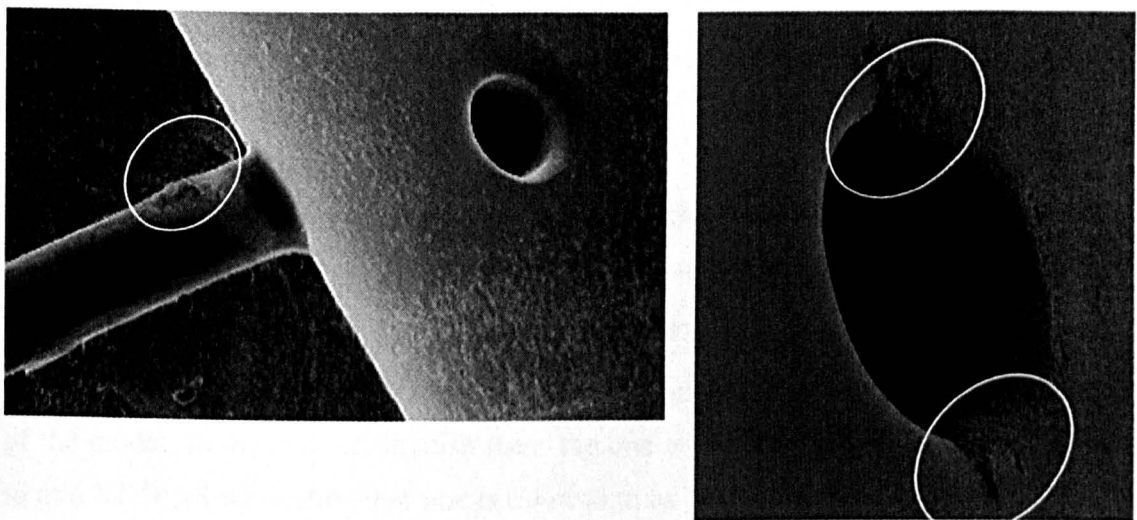
### **6.1 INTRODUCTION**

In the previous chapter the method was validated by several inverse design cases both using 2-D and 3-D geometries. In this chapter the discussion is mainly focused in the application of the developed method for producing optimised diesel nozzle shapes. For the purpose of optimising the geometries the second cost function is used, aiming to the maximisation of the discharge coefficient, which at the same time corresponds to geometries which have controlled or eliminated cavitation inception in the hole entrance. This is identified in the steady state mode by reduction of the volume of negative pressure appearing in the hole entrance. Several test cases, both 2-D and 3-D, are investigated in this chapter. Parametric studies of the effect of the different design parameters on the optimum geometry are presented. The geometries used are either constructed for the model testing or have been made available as part of recent and ongoing research of the group on cavitation.

### **6.2 OVERVIEW OF TEST CASES INVESTIGATED**

Cavitation appears in the nozzle in two forms. The first type initiates inside the orifice at the top of the hole entrance. It is formed due to flow separation at the corner that causes local pressure drop. Sometimes, cavitation erosion appears in this area. In a single-phase flow calculation this area can be correlated with the area where the calculated pressure takes negative values. The second form of cavitation is referred to as string cavitation. The root of this two-phase flow effect is still not fully documented. Some scenarios have been proposed in the literature. Relevant experimental work of the group published recently in [194], suggests vapour entrapment into the core of transient vortices forming inside the sac volume upstream of the injection hole. Evidence of the erosion damage caused by the two forms of cavitation can be seen in Figure 6.1, taken from [222]. The left hand-side image is due to geometrically-induced cavitation while the right-hand

side is believed to be due to string cavitation. Although the effect of cavitation on the spray and the overall engine performance is still under the microscope, the previously described results of its formation lead to the necessity of controlling the cavitation inception. During this study only the reduction of the geometrically-induced cavitation has been considered.



**Figure 6.1: Details of erosion in nozzle due to (left) hole cavitation and (right) string cavitation**

As it has been clear from the cost function definitions of section 3.4, the approach of the cost function (3.3), used for the validation is really restrictive because of the fact that the target pressure distribution must be known. For the application of the developed method in cavitation control the second cost function has been used. The cost function expression (3.5) is repeated here to facilitate the reader:

$$I_c = \frac{1}{2} (\bar{C}_d - C_d)^2 \quad (3.5)$$

where  $\bar{C}_d$  is the desired value of the  $C_d$  that we want to achieve. As it has been mentioned in section 3.2, although we seek to maximise the  $C_d$ , the problem is set up again as a minimisation problem by subtracting the current  $C_d$  value from the target  $\bar{C}_d$  one. This approach is based on the fact that cavitation is reduced or even eliminated as the geometric discharge coefficient ( $C_d$ ) increases. In this way cavitation inception can be “controlled” without actually solving for the multi-flow problem. It has to be noted that this is not an inherent disadvantage of the method adopted here, but simply a step not performed during the present investigation.

This approach is again an inverse design case but now the target geometry is not a known one. In most of the test cases in this chapter the cost function is used as an optimisation one where the user wishes to achieve the maximum  $C_d$  possible. For this reason the value of  $\bar{C}_d$  is set equal to 1; this is a non-feasible idealistic value that cannot be achieved. However, in some parametric studies where the effect of the different parameters is investigated the target  $\bar{C}_d$  is set to take actual feasible values for study reasons. In some cases constraint of the different parameter was essential so the design produced does not violate the manufacturer's standards.

In the next sections several geometries are tested using the aforementioned cost function. All the geometries are subjected to different tests. The effect of different target discharge coefficients  $\bar{C}_d$  in each parameter is investigated using the 2-D axisymmetric geometry. Two multi-hole VCO nozzles were used in different cases for the application of the model. In order to distinguish them the one which has no sac volume is referred to as a VCO NS while the other one is referred to as VCO WS. For the VCO NS nozzle the effect of different parameters is investigated in the  $C_d$  maximisation case. For the VCO WS nozzle the effect of different parameters is investigated as well as the effect of different needle lifts. Finally the effect of geometric constraint is depicted in the mini-sac nozzle. A description of the optimisation test cases investigated can be seen in Table 6.1 where the boundary conditions and the initial calculated discharge coefficient are depicted.

NOZZLE:	P <sub>INJ</sub>	P <sub>BACK</sub>	Initial C <sub>d</sub>
2-D axisymmetric	1000bar	20bar	0.81
6-Hole VCO NS	1000bar	20bar	0.76
6-Hole VCO WS	1300bar	100bar	0.78 (250µm lift) 0.16 (20 µm lift)
mini-sac	1600bar	160bar	0.84

**Table 6.1: Description of optimisation test cases**

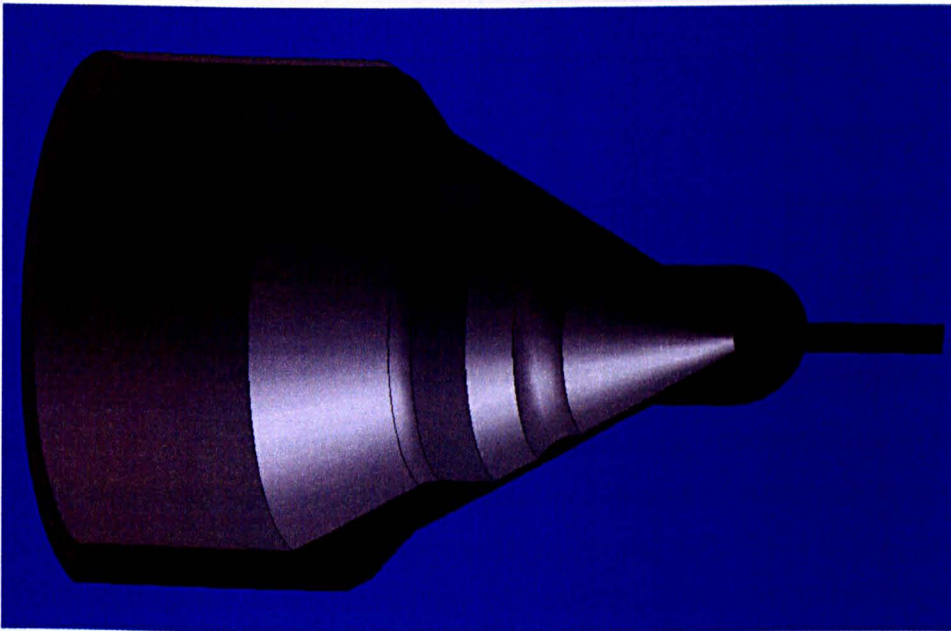
All four geometries are tested in high needle lift conditions apart from the VCO nozzle which is tested in both high and low needle lift. This choice was made because the discharge coefficient is usually specified at high needle lifts where there is no significant



pressure drop at the needle seat area and also because the grid modification tool works better for such cases.

### 6.3 AXISYMMETRIC SINGLE-HOLE NOZZLE

The same I-Level axisymmetric nozzle geometry which was used for the parametric studies of the previous chapter and was described in section 5.2 has been used here as well. More specifically, this geometry is the axisymmetric single-hole nozzle (Figure 5.1) that was constructed from a transparent acrylic material for purposes of comparison between experimental and computational data [216]. In Figure 6.2 a cross-section of the 3-D model of the sharp-edge axisymmetric single-hole Bosch nozzle is presented, revealing the needle and internal sac volume geometry.



**Figure 6.2: Half CAD model of axisymmetric 0%HE nozzle for I-Level project**

The initial geometry, diesel fuel used and flow conditions are the same as they have been described in detail in section 5.2. In the following sections the geometry is optimised with respect to the maximisation of the  $C_d$  and the effect of different design parameters in the optimisation process is investigated.

#### 6.3.1 Parametric study of the effect of the design parameters

Two design parameters have been used for the parameterisation of the 2-dimensional geometry. The effect of these parameters in the inverse design case with three target discharge coefficients is investigated in this section. First the effect of the conicity and



then the effect of the radius of curvature are discussed as they have been used to parameterise the 2-D axisymmetric nozzle according to the model described in section 3.5.

Effect of conicity

Three test cases are used to investigate the effect of the conicity in the inverse design process using different target discharge coefficients. These test cases are shown in detail in Table 6.2. The initial and target  $C_d$  are depicted as well as the predicted  $C_d$  using the optimisation model. In the same table the predicted percentage of increase from the initial conicity values are included. The initial geometry is cylindrical or has 0% conicity.

	Case1	Case2	Case3
Initial $C_d$	0.81	0.81	0.81
Target $C_d$	1.	0.93	0.88
Predicted $C_d$	0.92	0.90	0.88
Conicity	17%	14%	10%

Table 6.2: Cases and results for the effect of conicity

Calculation starts from an assumed initial geometry, corresponding to the dashed line (most cavitating design), as it is shown in Figure 6.3. After convergence, the model predicts the new geometries corresponding to the solid lines in the same Figure 6.3, which corresponds to nozzles having a  $C_d$  as close as possible to the targeted values.

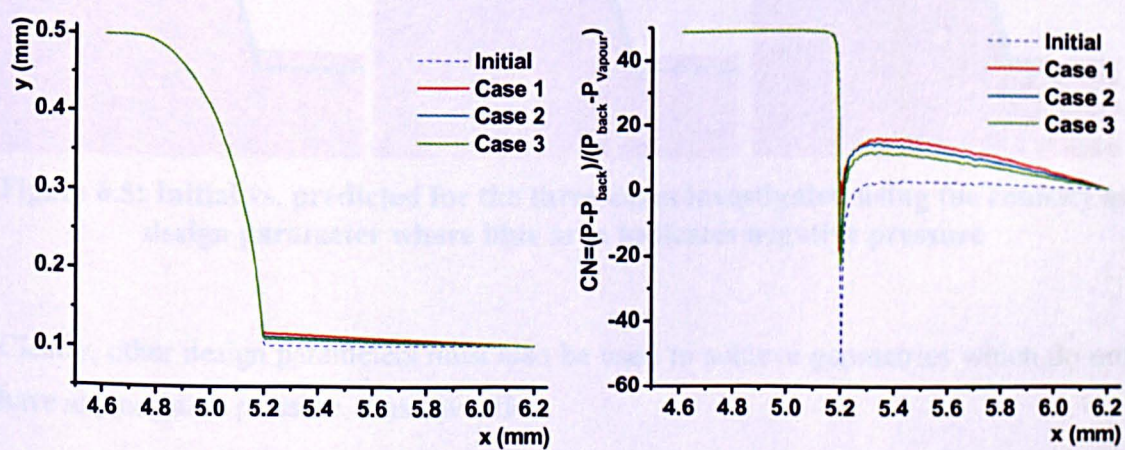
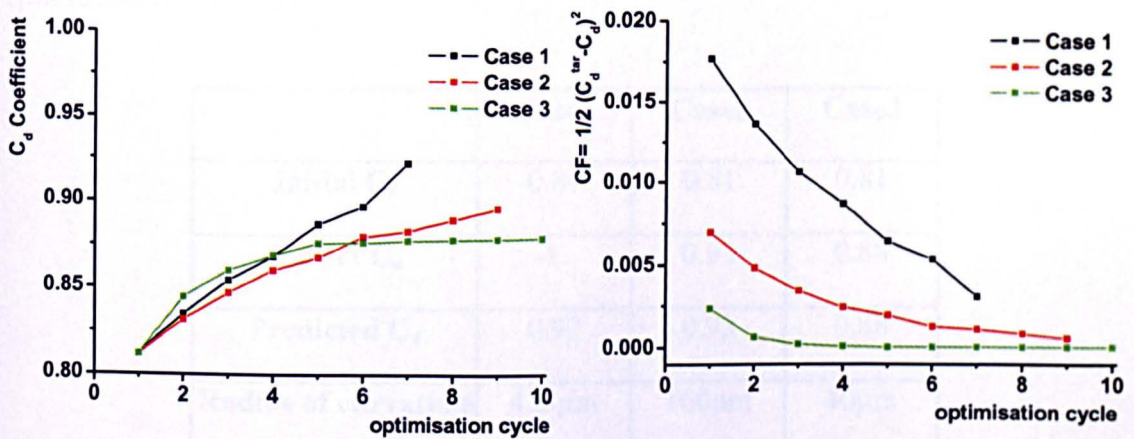


Figure 6.3: Initial and optimised boundary of the axisymmetric nozzle for the three cases using conicity as design parameter

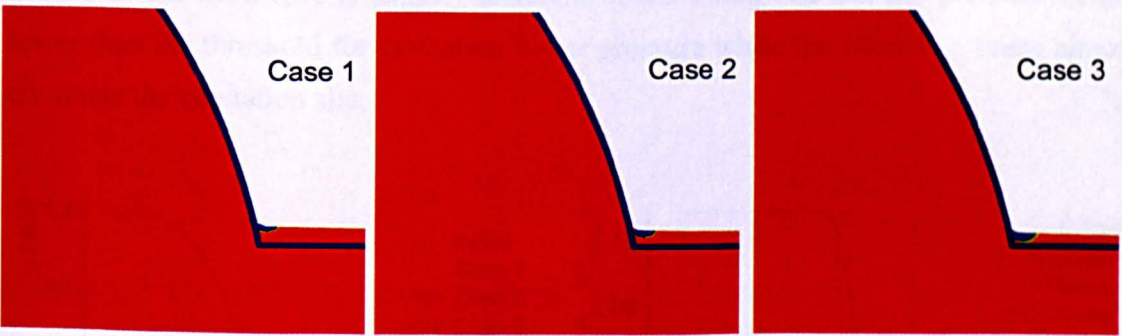


As it is clear from Table 6.2 and Figure 6.4, using just the conicity as design parameter the target  $C_d$  cannot be achieved in all three cases. This is an obvious result for the first case since the target  $\bar{C}_d$  is set equal to 1 which is not a feasible value. But the same problem appears also in the second case where  $\bar{C}_d = 0.93$ . Despite of the fact that using just this parameter there is no convergence in the third case the model converges in just 5 iterations (Figure 6.4).



**Figure 6.4: History of the discharge coefficient and cost function for the three test cases where only the conicity is used as design parameter**

The contour of the predicted geometries for the three cases investigated are presented in Figure 6.5 and compared to the initial geometry. The red area shows positive pressure values while the blue indicates the negative pressure areas.



**Figure 6.5: Initial vs. predicted for the three cases investigated using the conicity as design parameter where blue area indicates negative pressure**

Clearly, other design parameters must also be used to achieve geometries which do not have any negative pressure, thus cavitation.



Effect of radius of curvature

The same three test cases are used to also investigate the effect of the radius of curvature in the inverse design process using the same three target discharge coefficients. These test cases are shown in detail in Table 6.23. The initial and target  $C_d$  are depicted as well as the predicted  $C_d$  using the optimisation model. In the same table the predicted radius of curvature values are included. The initial geometry is almost sharp with only  $1\mu\text{m}$  radius of curvature.

	Case1	Case2	Case3
Initial $C_d$	0.81	0.81	0.81
Target $C_d$	1.	0.93	0.88
Predicted $C_d$	0.97	0.93	0.88
Radius of curvature	425 $\mu\text{m}$	160 $\mu\text{m}$	40 $\mu\text{m}$

Table 6.3: Cases and results for the effect of radius of curvature

Again the design process starts from the dashed line shown in Figure 6.6 and converges to the predicted geometries depicted with the solid lines and their corresponding pressure distributions which are shown in the same Figure 6.6. The predicted designs correspond again to nozzles having a  $C_d$  as close as possible to the targeted values. The geometry of the third case is slightly different of the initial one but has pressure values lower than the threshold for cavitation lower pressure while the other two cases almost eliminate the cavitation site.

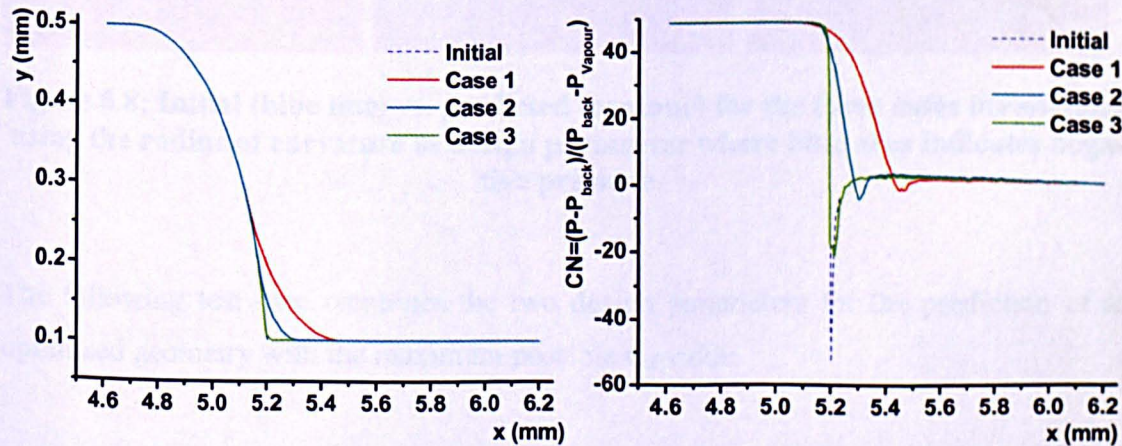
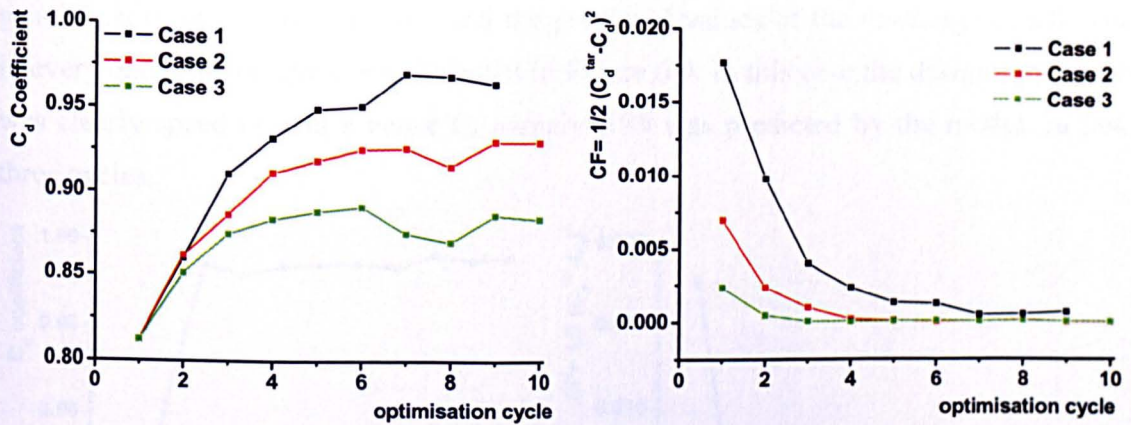


Figure 6.6: Initial and optimised boundary of the axisymmetric nozzle for the three cases using radius of curvature as design parameter

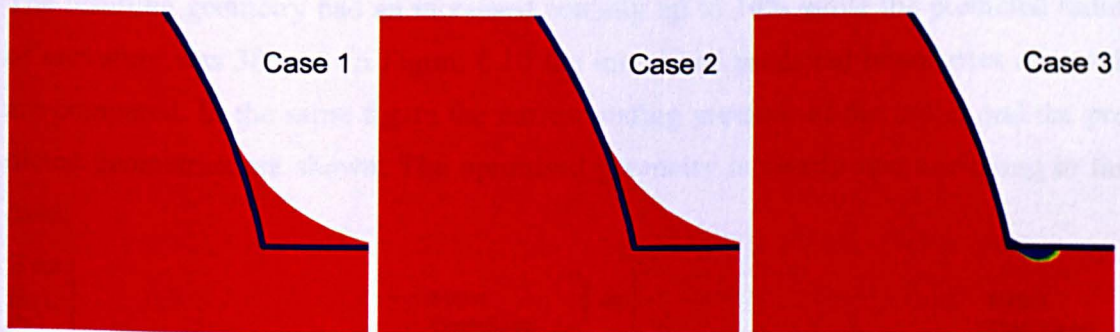


This time the model managed to predict exactly the two inverse cases 2 and 3 as it is obvious from Table 6.3 and from the convergence history shown in Figure 6.7. The model reached a maximum  $C_d$  value of 0.97 in less than eight iterations, as can be seen in the same Figure 6.7.



**Figure 6.7: History of the discharge coefficient and cost function for the three test cases where only the radius of curvature is used as design parameter**

In Figure 6.8 the predicted contour is presented and compared with the initial geometry. From the red colour it is obvious that the results of cases 1 and 2 are not expected to result to cavitation inception. The third resulting geometry is cavitating because of the choice of the target discharge coefficient.



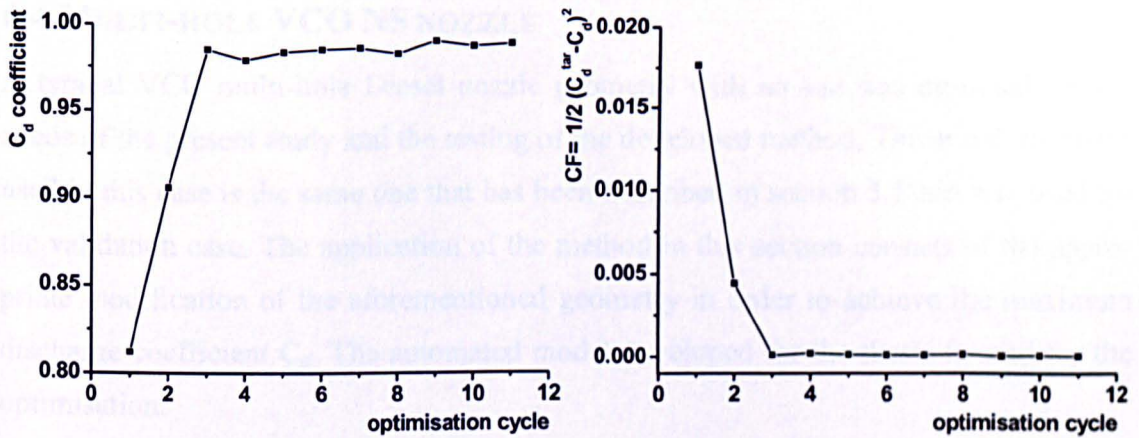
**Figure 6.8: Initial (blue line) vs. predicted (contour) for the three cases investigated using the radius of curvature as design parameter where blue area indicates negative pressure**

The following test case combines the two design parameters for the prediction of an optimised geometry with the maximum possible  $C_d$  value.



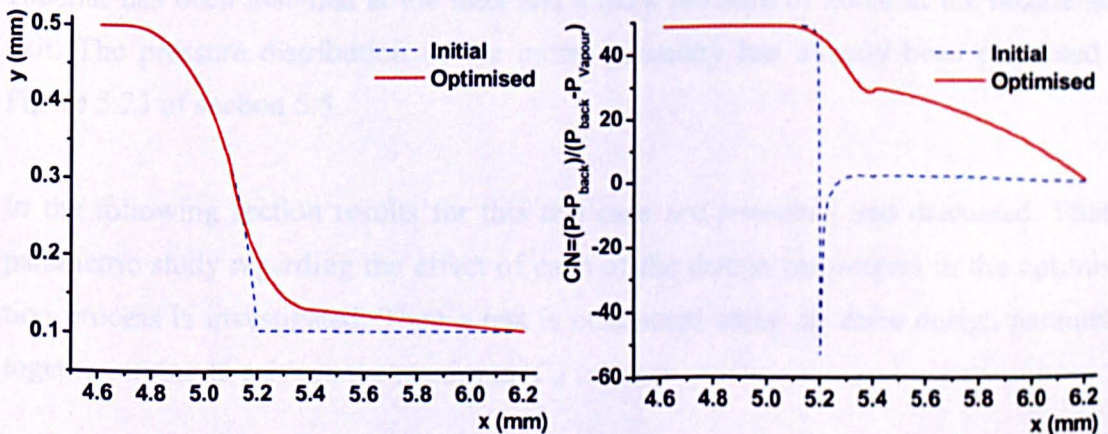
### 6.3.2 Discharge coefficient maximisation of the 2-D axisymmetric single-hole nozzle

The predescribed optimisation procedure was repeated in this case where both parameters are used for the design of the 2-D axisymmetric single-hole nozzle. The convergence history of the cost function and the predicted values of the discharge coefficient in every optimisation cycle are presented in Figure 6.9. In this case the design procedure was clearly speed up and a better  $C_d$  namely 0.99 was predicted by the model, in just three cycles.



**Figure 6.9: History of the discharge coefficient and cost function for the optimisation case**

The resulting geometry had an increased conicity up to 14% while the predicted radius of curvature was  $380\mu\text{m}$ . In Figure 6.10 the initial and predicted boundaries of nozzle are compared. In the same figure the corresponding pressure of the initial and the predicted geometries are shown. The optimised geometry is clearly non cavitating in this case.



**Figure 6.10: Initial vs. optimised boundary of the axisymmetric nozzle after convergence of the cost function**

Clearly, by using both parameters the resulting geometry can achieve the largest  $C_d$  values and the cavitation inception can be completely eliminated. The design parameters in this case needed less modification to predict a larger  $C_d$  value than the extreme cases of each parameter by itself. This is a very convenient result since usually the manufacturers are seeking for improvement in the performance resulting by slight geometry modification. The model also converges much faster using the freedom given to it by the dimension in the search space.

## 6.4 MULTI-HOLE VCO NS NOZZLE

A typical VCO multi-hole Diesel nozzle geometry with no sac was designed for the needs of the present study and the testing of the developed method. The initial geometry used in this case is the same one that has been described in section 5.5 and was used for the validation case. The application of the method in this section consists of the appropriate modification of the aforementioned geometry in order to achieve the maximum discharge coefficient  $C_d$ . The automated model developed for the thesis is used for the optimisation.

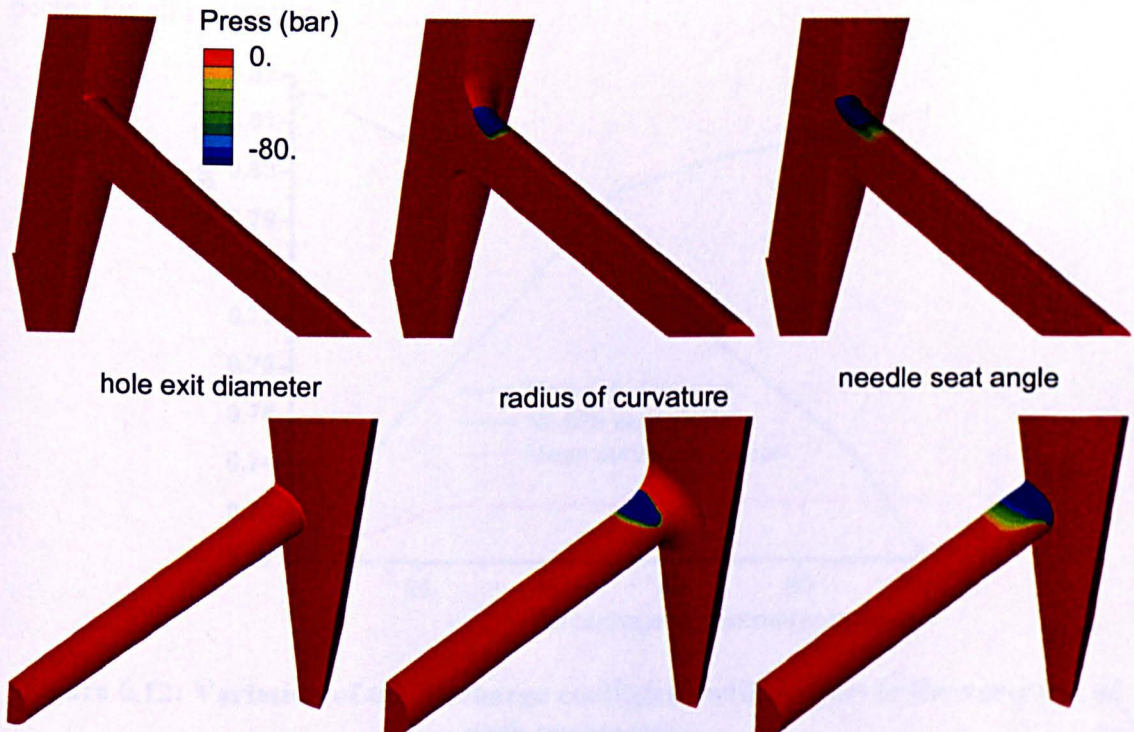
The specifications of the geometry are described in section 5.5 but are repeated here to facilitate the reader. The hole diameter of the nozzle is  $152\mu\text{m}$  and the hole length  $970\mu\text{m}$ . The geometry simulated corresponds to the half of one of the six holes, thus the computational domain corresponds to  $1/12^{\text{th}}$  of the whole nozzle design (details can be seen in Figure 5.21) and it consists of approximately 50,000 cells. Symmetry boundary conditions have been used at the left and side boundaries while an injection pressure of 1000bar has been assumed at the inlet and a back pressure of 20bar at the nozzle hole exit. The pressure distribution of the initial geometry has already been presented in Figure 5.23 of section 5.5.

In the following section results for this test case are presented and discussed. First a parametric study regarding the effect of each of the design parameters in the optimisation process is investigated. Then a test is conducted using all three design parameter together, trying to achieve the maximum  $C_d$  value.



### 6.4.1 Effect of different parameters

Tests have been conducted for all three different design parameters to investigate the effect of their perturbation in the value of the discharge coefficient. The ability of the method to predict nozzles without cavitation by changing only one parameter each time is also investigated. In Figure 6.11 the resulting pressure distributions and geometries for each parameter are presented. The positive pressure is denoted with red while the contour shows the area of negative pressure in the predicted nozzles.

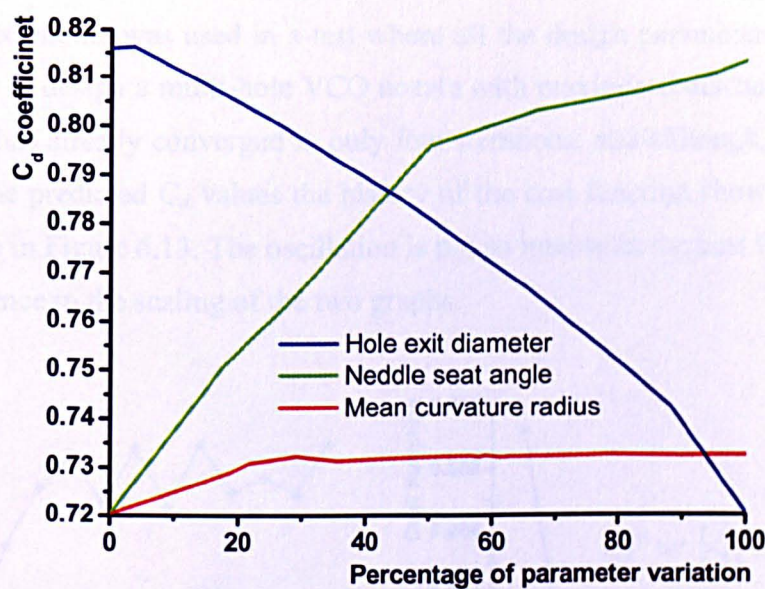


**Figure 6.11: Effect of different parameters of the VCO NS nozzle in the  $C_d$  maximisation case**

As it is obvious from Figure 6.11 changing only the conicity, a geometry with no cavitation inception can be produced. Nevertheless, this is the case for tapered nozzles where a more gradual pressure drop happens along the length of the hole which results to overall higher  $C_d$ . Concerning the other two parameters some more conclusions can be drawn. Modifying the radius of curvature leads to displacement of the negative pressure area further inside the hole compared to the pressure distribution of the initial geometry (Figure 5.23). Finally, the inclination angle when acting as the unique design variable does not make any significant difference in the cavitation intensity for this particular case tested here. This is not a general conclusion and it believed to be due to the shape of the sac of the nozzle and the position of the hole. Nevertheless, this design parameter is usually restricted within a small range of variation, as specified by the engine design.



The effect of each parameter variation on the discharge coefficient values is depicted in Figure 6.12. The x-axis represents the percentage of increase in each parameters variation. Moreover, 0% is considered the variation of the parameters in the initial design and 100% is the maximum predicted variation for each parameter. As it has been discussed in section 3.5 where the parameterisation scheme was described the radius of curvature varies for each *zone* of the hole surface. For this reason the mean/average value of these radiuses is used in Figure 6.12. The behaviour of the discharge coefficient is the expected for all parameters.



**Figure 6.12: Variation of the discharge coefficient with respect to the variation of each parameter**

In Table 6.4 the resulting  $C_d$  and the predicted value of each parameter are summarised, as well as the normalised volume of negative pressure (VNP) of the predicted geometries relative to that of the initial design.

	Cylindrical	Tapered	Curved	Inclined
$C_d$	0.72	0.82	0.81	0.73
Design parameter	0	5.2°	40.μm	1°
VNP	100%	0.5%	52%	91%

**Table 6.4: Summary of results for the effect of the design parameters in the Multi-hole VC NS nozzle optimisation**

Clearly, the tapered geometry is the one which can achieve the largest  $C_d$  with an angle of  $5.2^\circ$  compared to the cylindrical geometry and a 99.5% of decrease of the VNP is achieved. The radius of curvature can produce geometry with almost 50% of the VNP of the sharp one while the inclination angle does not really vary during the optimisation process. In the next paragraph all parameters are combined for the optimisation of the multi-hole nozzle.

#### 6.4.2 Discharge coefficient maximisation of multi-hole VCO NS nozzle

The developed model was used in a test where all the design parameters were free to vary in order to design a multi-hole VCO nozzle with maximised discharge coefficient. The method has already converged in only four iterations, and although, there is an oscillation in the predicted  $C_d$  values the history of the cost function shows convergence, as it is shown in Figure 6.13. The oscillation is not so intense in the cost function history due to difference in the scaling of the two graphs.

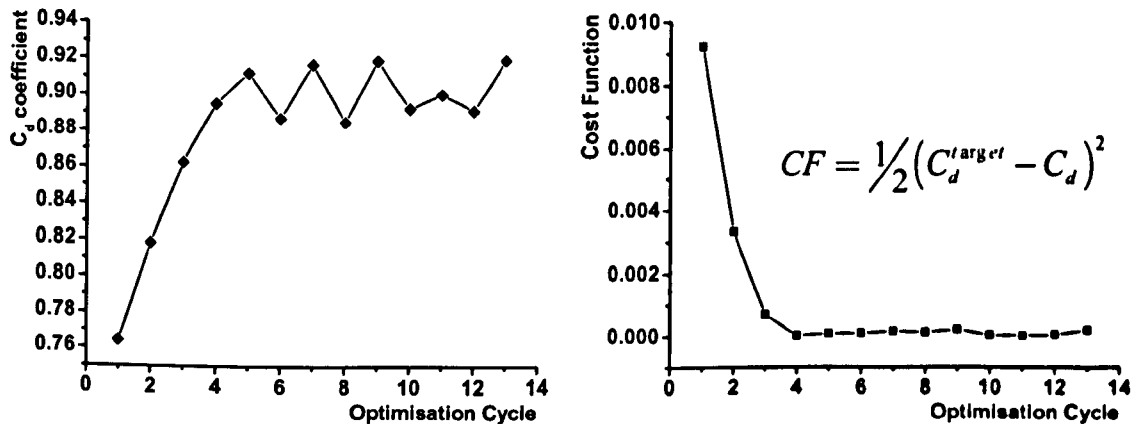
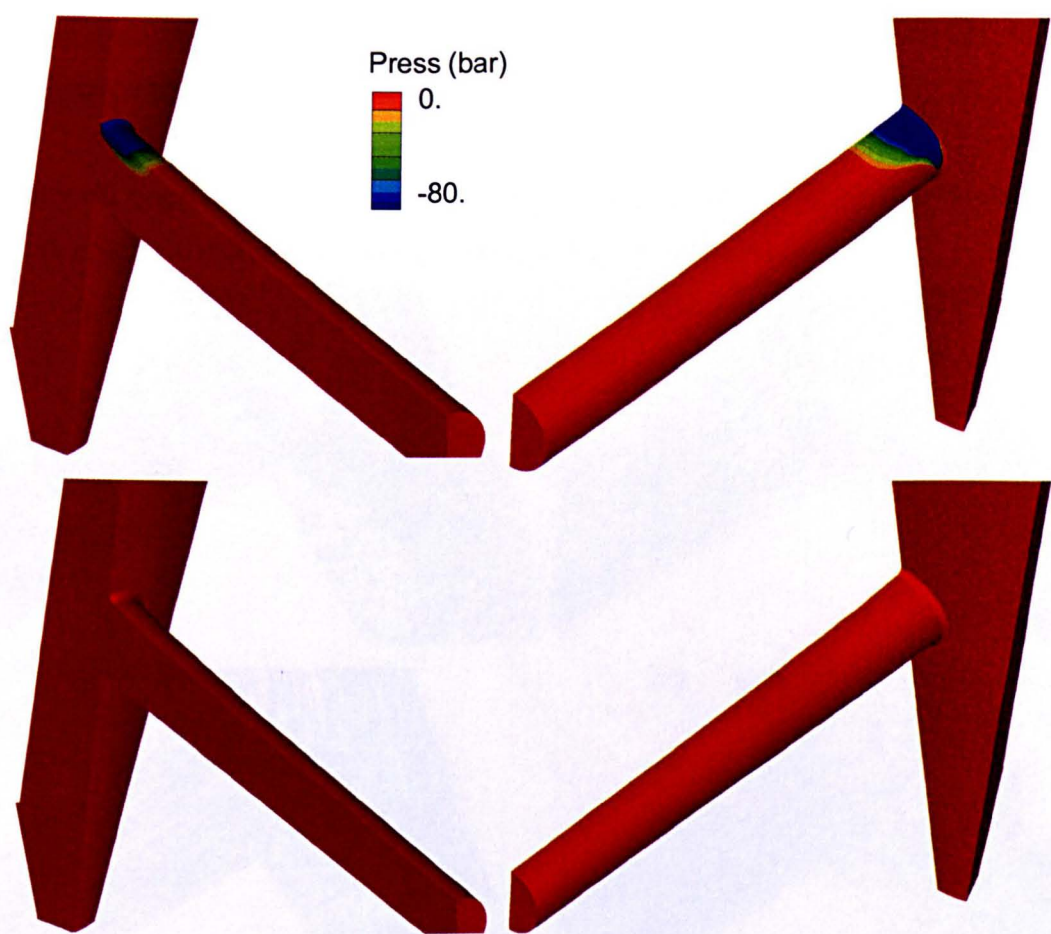


Figure 6.13: History of the discharge coefficient and cost function for the 6-hole VCO NS nozzle case

The resulting geometry appears to be non cavitating since the negative pressure values are eliminated as it is shown in Figure 6.14. On the same Figure 6.14 the initial and optimised pressure contour and geometry of the VCO nozzle after convergence of the cost function is presented. The positive pressure values are depicted with red colour while the other colours show the negative pressure contour with blue being the most intense. It is obvious that the model predicted a cavitation free geometry with maximised  $C_d$  by introducing small changes in the radius of curvature and the conicity while the inclination angle again, did not contribute to the overall result.





**Figure 6.14: Initial and optimised pressure contour of the VCO NS nozzle after convergence of the cost function**

The predicted values of the design parameters, the  $C_d$  and the VNP have been compared to the initial ones and they are summarised in Table 6.5.

	$C_d$	VNP	Radius of curvature	Conicity	Needle seat angle
Initial	0.76	100%	10.μm	0°	0°
Predicted	0.92	0%	12.5μm	2.4°	1°

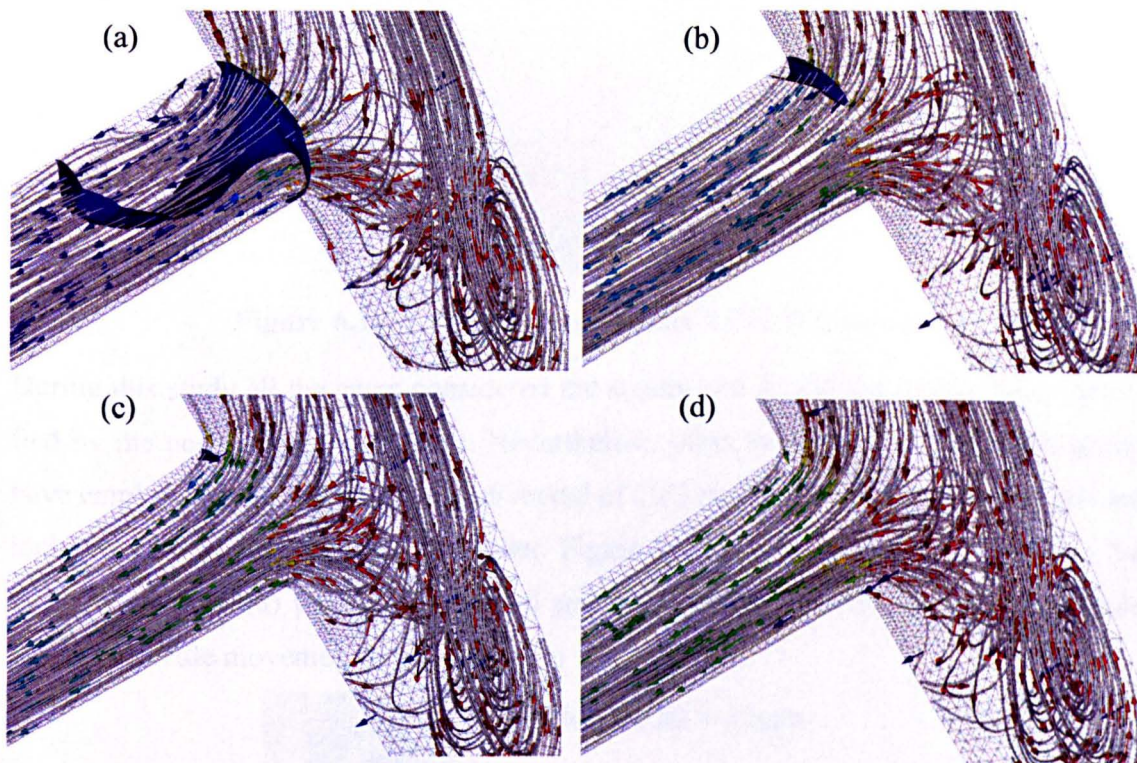
**Table 6.5: Results of the Multi-hole VCO NS nozzle after convergence of the cost function**

Another point that is noticeable from the results is that the predicted discharge coefficient in the case that all parameters are involved is the best prediction compared to the tests with each individual design parameter. Also in this case the parameters have to vary less to predict this optimum value compare to the case where each parameter varies individually. The starting  $C_d$  case is larger than in the parametric study because the



starting geometry in this case is not completely sharp since so as to resemble realistic initial geometry.

Finally, in Figure 6.15 the flow field and the iso-surface of negative pressures of the initial, predicted and two in-between iterations geometries are presented.



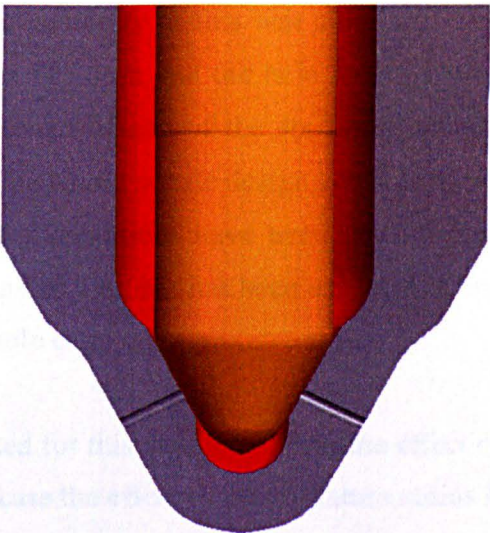
**Figure 6.15: Streamlines and iso-surface of negative pressure of the (a) initial, (b) third cycle, (c) tenth cycle and (d) predicted geometries for the Multi-hole VCO NS nozzle case**

The streamlines coming in the hole along with the iso-surfaces of the negative pressure area are depicted. In Figure 6.15 (a) and (b) the recirculation and separation producing the negative pressure respectively are shown. The flow is attached to the hole entrance in Figure 6.15 (c) due to the hole inlet curvature predicted by the model.

## 6.5 MULTI-HOLE VCO WS NOZZLE

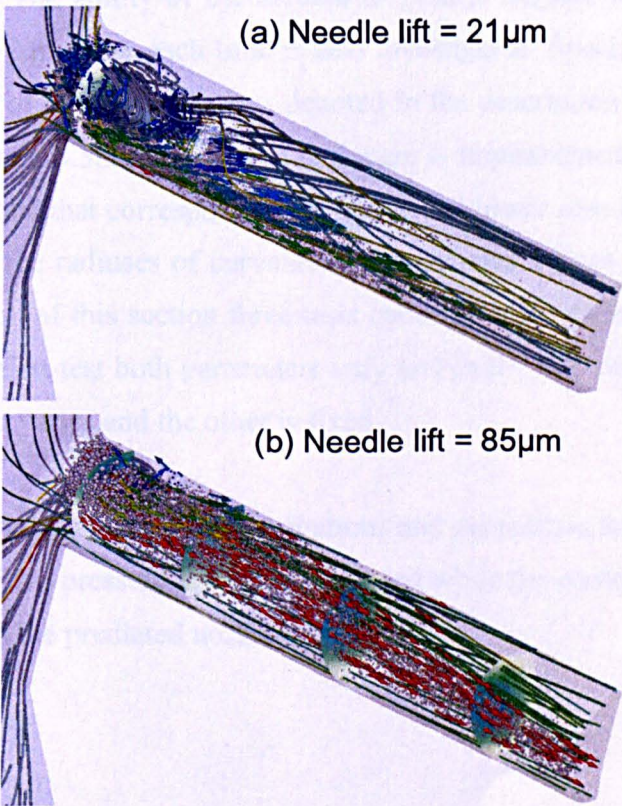
The model has also been employed for the optimisation of a multi-hole VCO Diesel nozzle. The geometry of Figure 6.16 has been provided by the manufacturer to the group for investigation purposes. Moreover, the cavitation effects of this nozzle and geometry modifications in order to reduce them are investigated. Both experimental and computational tests have been conducted in this geometry by the CFD group of City University but not within the context of this thesis.





**Figure 6.16: CAD model of 6-hole VCO WS nozzle**

During this study all the cases considered are steady and the cavitation has been identified by the negative pressure areas. Nevertheless, other member of the research group have employed the in-house cavitation model of [18] and some representative results are included here as reference for the reader. Figure 6.17 shows the cavitating flow in the VCO WS nozzle (a) when it is initiated and (b) when the cavitation reaches the hole exit. The needle movement is considered in this case.



**Figure 6.17: Results from unsteady cavitating flow when (a) cavitation is initiated and (b) when it reaches the exit of the hole for the VCO WS nozzle**

The specifications of the geometry as this was simulated are described next. The hole diameter of the nozzle is  $195.6\mu\text{m}$  and the hole length  $1\text{mm}$ . The geometry simulated corresponds again to the half of one of the six holes, thus the computational domain corresponds to  $1/12^{\text{th}}$  of the whole nozzle design and it consists of approximately 65,000 cells. Symmetry boundary conditions have been used at the left and side boundaries while an injection pressure of 1300bar has been assumed at the inlet and a back pressure of 100bar at the nozzle hole exit.

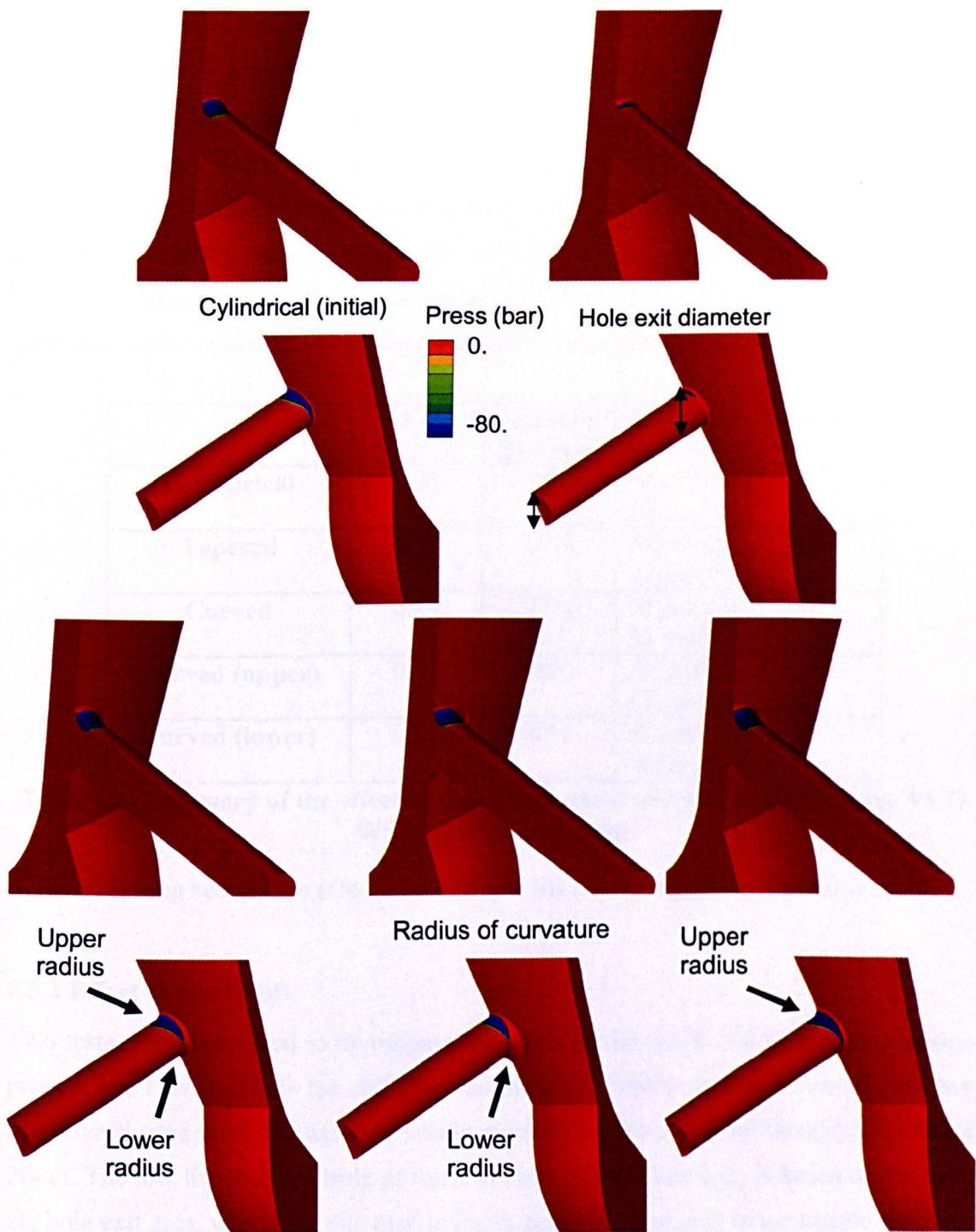
Two cases are investigated for this geometry. First the effect of the different parameters is studied, where in this case the effect of the curvature radius is explored in more detail. The effect of the needle lift is explored for this case as well. Significant results are drawn from the aforementioned tests which are discussed in the following sections.

#### 6.5.1 Effect of design parameters

Tests for each parameter are conducted for this geometry in the way they were employed in the geometry of section 1916.4.1. The model was applied in each individual design parameters to investigate the effect of their perturbation in the value of the discharge coefficient. The ability of the method to predict nozzles without cavitation by changing only one parameter each time is also investigated. Special attention has been given in the radius of curvature. As it is denoted in the description of the parameterisation scheme in section 3.5.2 the radius of curvature is implemented in the design by the use of two parameters that correspond to the upper and lower *zones* of the hole. The in-between values of the radiuses of curvature are interpolated from the two given ones. For the investigation of this section three tests concerning the radius of curvature were carried out. In the first test both parameters vary and in the two only the lower or only the upper parameter varies and the other is fixed.

In Figure 6.11 the resulting pressure distributions and geometries for each parameter are presented. The positive pressure is denoted with red while the contour shows the area of negative pressure in the predicted nozzles.





**Figure 6.18: Effect of different parameters of the VCO WS nozzle in the  $C_d$  maximisation case**

As it is clear from Figure 6.18 only the conicity almost eliminates the cavitation in the nozzle. An unexpected result for the three curvature cases is that the best behaviour was observed when the lower curvature varies; something that was not the case in previous tests (section 5.5). In this case the performance of the nozzle is mainly affected by the lower curvature where in the aforementioned validation case that parameter did not play

a significant part in the results. The author believes that this is due to the existence of sac volume in this geometry. Another observation was that in some cases the discharge coefficient was increased while the VNP did not decrease. In those cases the area of negative pressure was transferred further inside the hole of the orifice and its intensity was weakened. Because of that the negative pressure in this case was measured by the average between the percentage of VNP and of the difference of the maximum pressure between the predicted and the initial design. In Table 6.6 all these results concerning the behaviour of the model for each design parameter are summarised.

	$C_d$	Negative pressure	Radius of curvature	Conicity
<b>Cylindrical</b>	0.78	100%	20.μm(up) 12.μm(lr)	0°
<b>Tapered</b>	0.87	13%	20.μm(up) 12.μm(lr)	4.3°
<b>Curved</b>	0.84	92%	36.μm(up) 31.μm(lr)	0°
<b>Curved (upper)</b>	0.83	96%	38.μm(up) 12.μm(lr)	0°
<b>Curved (lower)</b>	0.85	80%	20.μm(up) 78.5μm(lr)	0°

**Table 6.6: Summary of the effect of the design parameters in the Multi-hole VCO WS nozzle optimisation**

In the following section the effect of the needle lift is investigated on the same nozzle.

### 6.5.2 Effect of needle lift

Two tests were conducted to investigate the effect of the needle lift in the optimisation process and how it affects the ability of the model to predict optimum results. The two lifts investigated were the extreme needle placements for this case namely 250μm and 20μm. The low lift case is a little peculiar because the predicted  $C_d$  is based on the nozzle hole exit area, which for this case is larger than the flow area at the needle seat. For the purposes of making a *fair* comparison the two cases a target  $\bar{C}_d = 0.25$  was used in the low lift case. So both cases converged in approximately 30 to 35 iterations giving a very good estimation of the target  $\bar{C}_d$  as it is clear from Figure 6.19. Results showed that the low lift case starting from a  $C_d$  of 0.16 with small variation in the curvature and the conicity achieves a cavitation free solution while for the high lift case the significantly larger variation in the curvature is necessary.

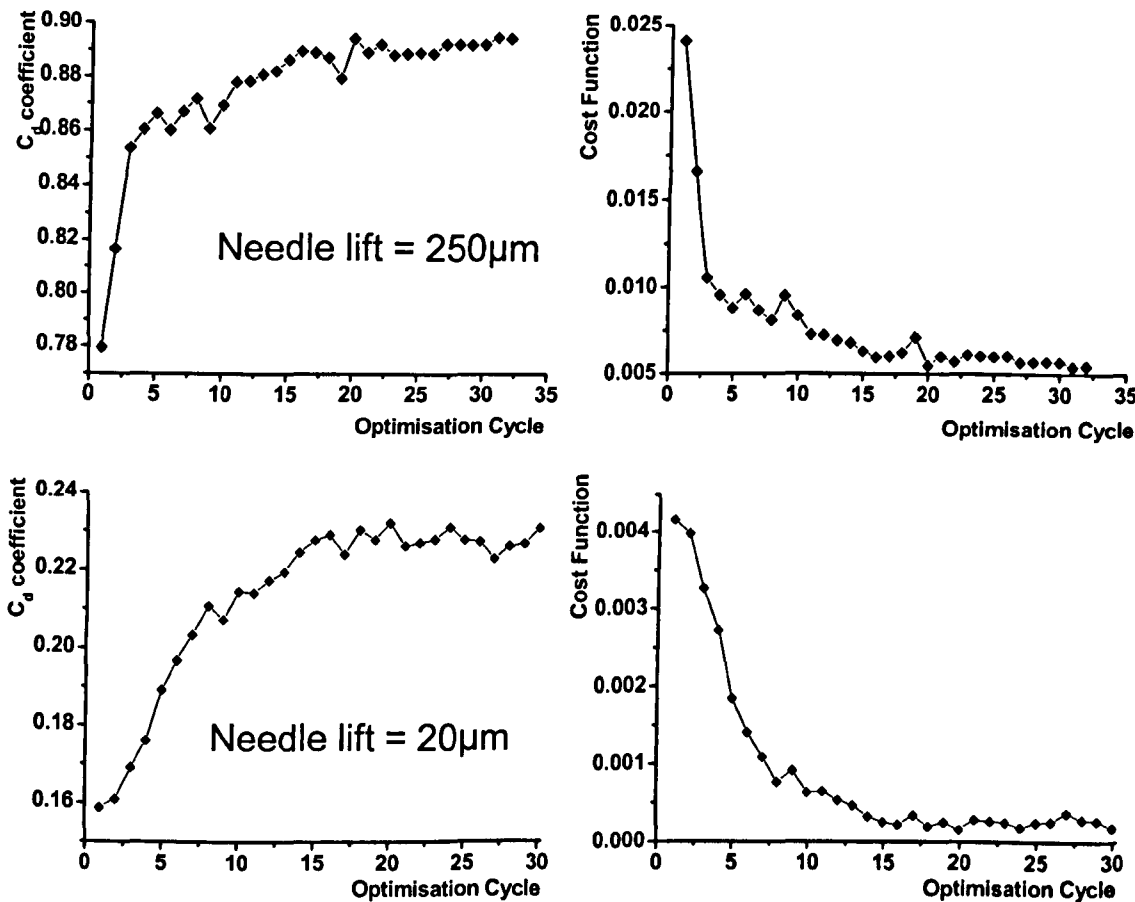


Figure 6.19: History of the discharge coefficient and cost function for the VCO WS nozzle in two different needle lifts

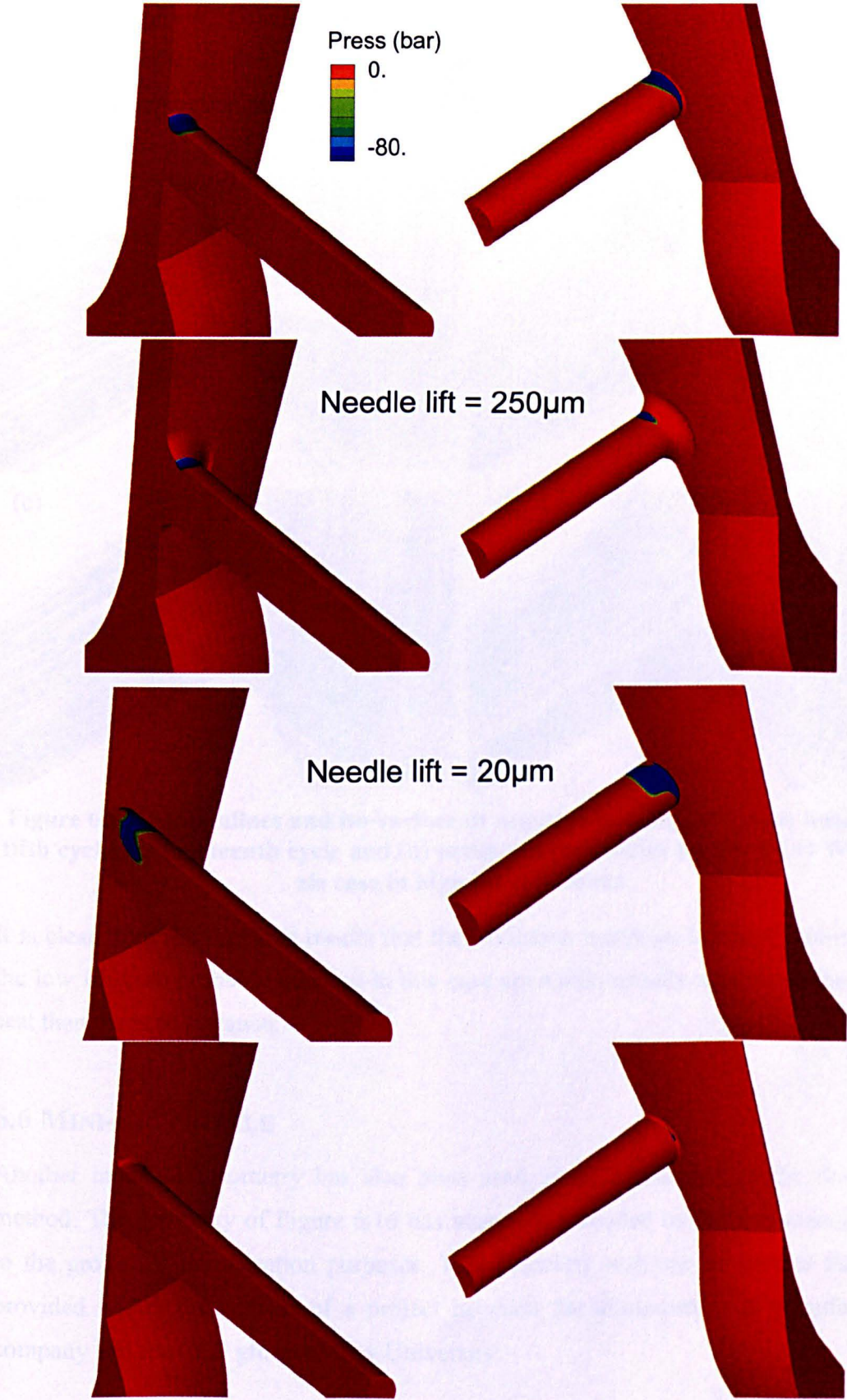
All these results concerning the effect of the needle lift in the optimisation of the VCO WS nozzle are summarised in Table 6.7.

	$C_d$	VNP	Radius of curvature	Conicity	Needle seat angle
Initial (20µm)	0.16	100%	10.µm(up) 8.µm(lr)	0°	0°
Predicted (20µm)	0.23	0%	13.µm(up) 9.µm(lr)	5°	0°
Initial (250µm)	0.78	100%	20.µm(up) 12.µm(lr)	0°	0°
Predicted (250µm)	0.895	3%	90.µm(up) 77.µm(lr)	4.8°	0.4°

Table 6.7: Summary of the effect of the needle lift in the Multi-hole VCO WS nozzle optimisation

In Figure 6.20 the initial and predicted geometries of both the high and low lift case are cross referenced where the previously discussed attributes can be visualised.

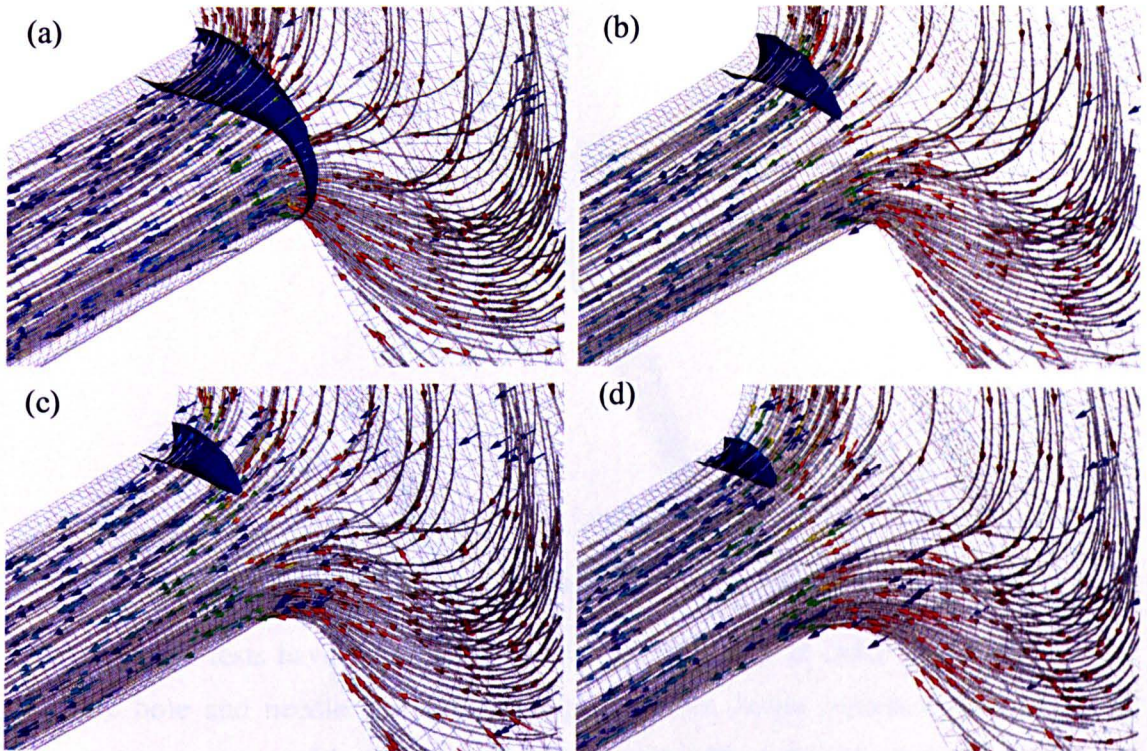




**Figure 6.20: Initial and optimised pressure contour of the VCO WS nozzle in 250µm (upper) and 20µm (lower) needle lift after convergence of the cost function**



Finally in Figure 6.21 the streamlines and the iso-surfaces of the negative pressure in the initial, predicted and two in-between iterations for the high lift case are presented and also the separation that causes the cavitation inception in the hole entrance.



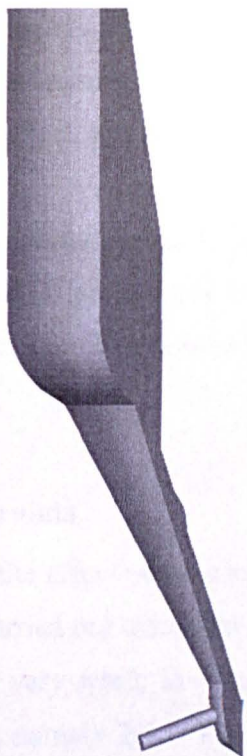
**Figure 6.21: Streamlines and iso-surface of negative pressure of the (a) initial, (b) fifth cycle, (c) fourteenth cycle and (d) predicted geometries for the VCO WS nozzle case in high lift conditions**

It is clear from the previous results that the cavitation inception is easier eliminated in the low lift case probably because in this case cavitation usually appears in the needle seat than the hole entrance.

## 6.6 MINI-SAC NOZZLE

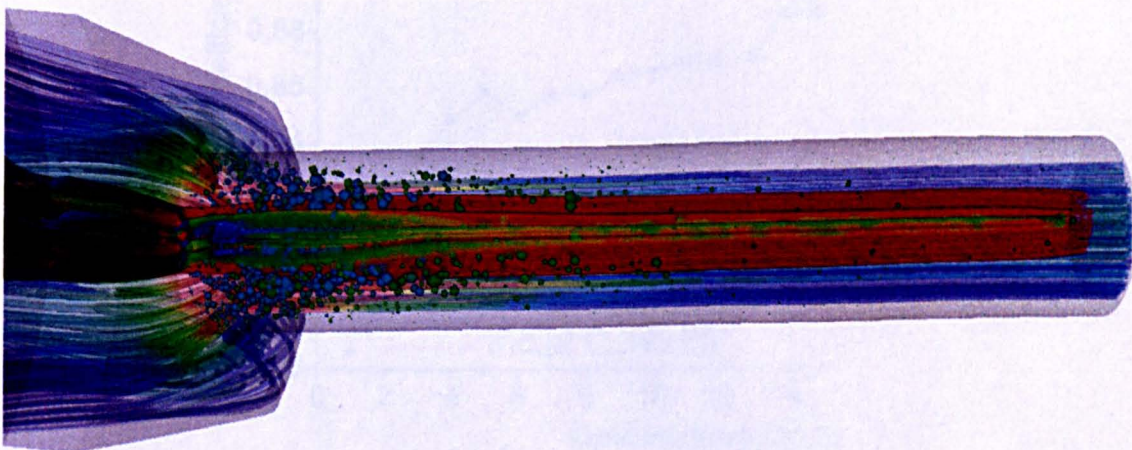
Another industrial geometry has also been used as an application to the developed method. The geometry of Figure 6.16 has been also provided by another manufacturer to the group for investigation purposes. This geometry was one of several that were provided within the context of a project between the aforementioned manufacturing company and the CFD group of City University.





**Figure 6.22: Half of one hole of the mini-sac 6-hole nozzle geometry**

Computational tests have been conducted in this geometry in order to identify and investigate hole and needle seat cavitation phenomena. Some representative results of these tests are presented in Figure 6.23, taken from [18]. Although, these kind of flows are not included in the present research it is included here as reference for the reader for a better comprehension of the multi-phase flow in such nozzles.



**Figure 6.23: Results for cavitating flow the mini-sac multi-hole nozzle**

The specifications of this geometry are: the hole diameter of the nozzle is  $120.\mu\text{m}$  and the hole length  $623.\mu\text{m}$ . The geometry simulated corresponds again to the half of one of the six holes, thus the computational domain corresponds to  $1/12^{\text{th}}$  of the whole nozzle



design and it consists of approximately 60,000 cells. Symmetry boundary conditions have been used at the left and side boundaries while an injection pressure of 1600bar has been assumed at the inlet and a back pressure of 160bar at the nozzle hole exit.

The effect of the geometrical constraints is investigated for this geometry. Two tests have been conducted where in one all parameters are free to vary and in the other the conicity has an upper boundary of  $2.5^{\circ}$ . The results which are drawn from this test are discussed in the following section.

6.6.1 Effect of geometrical constraints

For the purposes of investigating the effect of geometrical constraints on the developed method two different tests were carried out using the mini-sac nozzle. In the first test all the design parameters are free to vary while in the second case an upper limit in the values of the conicity is enforced, namely  $2.5^{\circ}$ . The problem again consisted of maximising the discharge coefficient and a prediction was achieved for both cases in about 15 iterations. The convergence history of the discharge coefficient for the unconstraint case is shown in Figure 6.24 where the initial and final predictions of discharge coefficient are depicted.

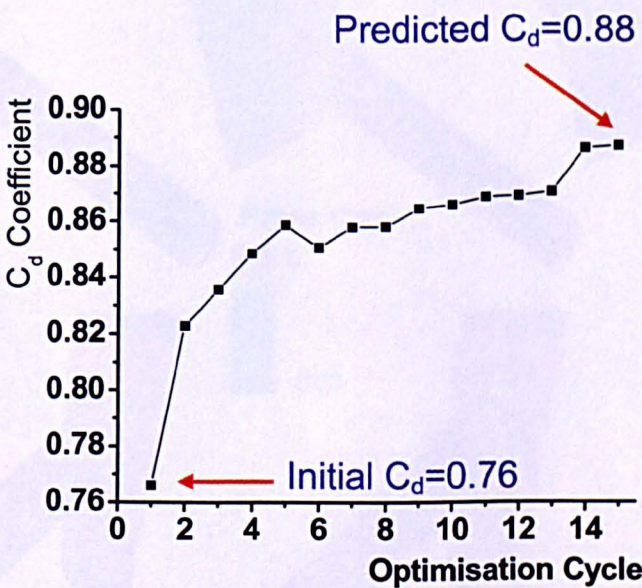


Figure 6.24: History of the discharge coefficient for the mini-sac nozzle case without constraints



The corresponding results are summarised in Table 6.8 where it is obvious that the case with the constraint conicity resulted in a geometry with smaller  $C_d$ .

	$C_d$	VNP	Radius of curvature	Conicity	Needle seat angle
Initial	0.76	100%	10.μm	0°	0°
Predicted (constraint conicity)	0.86	15%	16.6μm(up) 13.7μm(lr)	2.4°	0.6°
Predicted (free parameters)	0.88	0%	14.μm(up) 10.4μm(lr)	5.34°	0.7°

Table 6.8: Results of the mini-sac nozzle for constraint and unconstraint optimisation

The inclination angle again in this case did not influence the outcome of the optimisation. Another noticeable result is the fact that the constraint of the conicity influenced the modification of the radius of curvature. Moreover, the optimisation model modified the radius of curvature more in order to predict a nozzle with larger  $C_d$ . Clearly from the summary of the results in Table 6.8 and Figure 6.25 the model did not achieve to eliminate the cavitation inception in the constraint case.

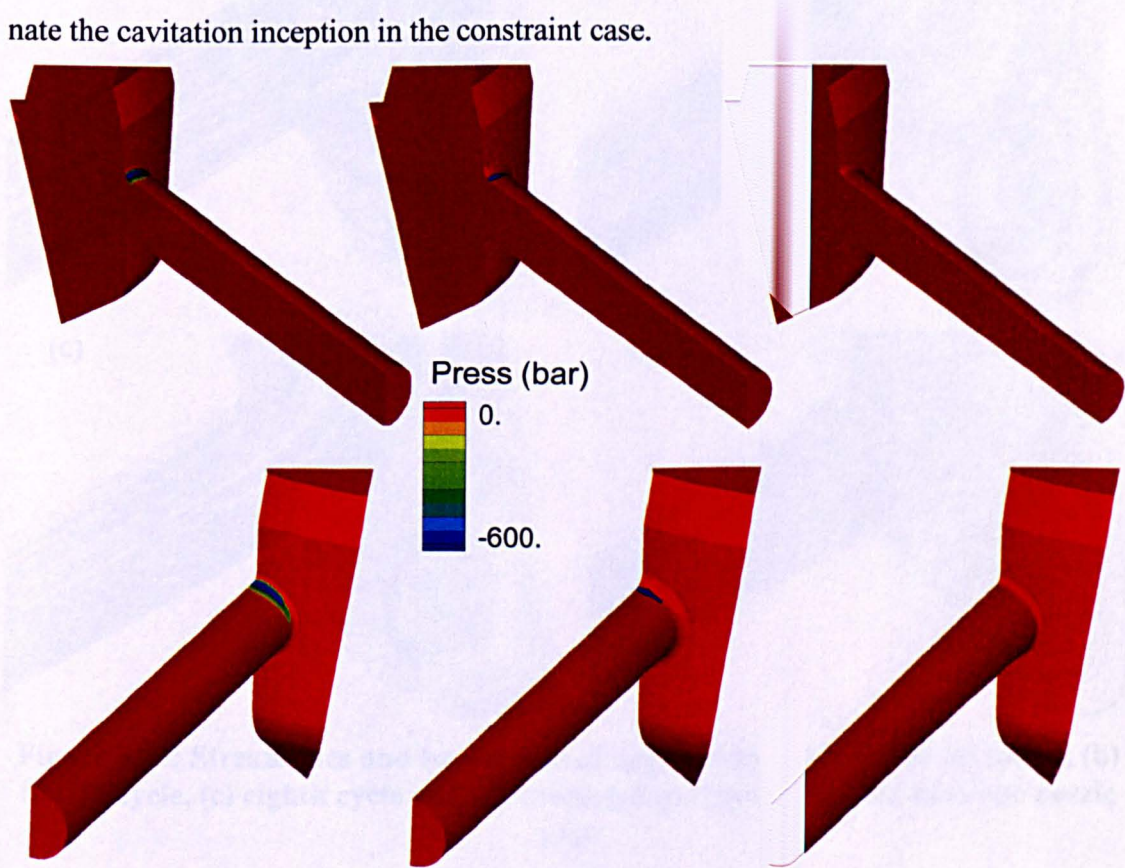
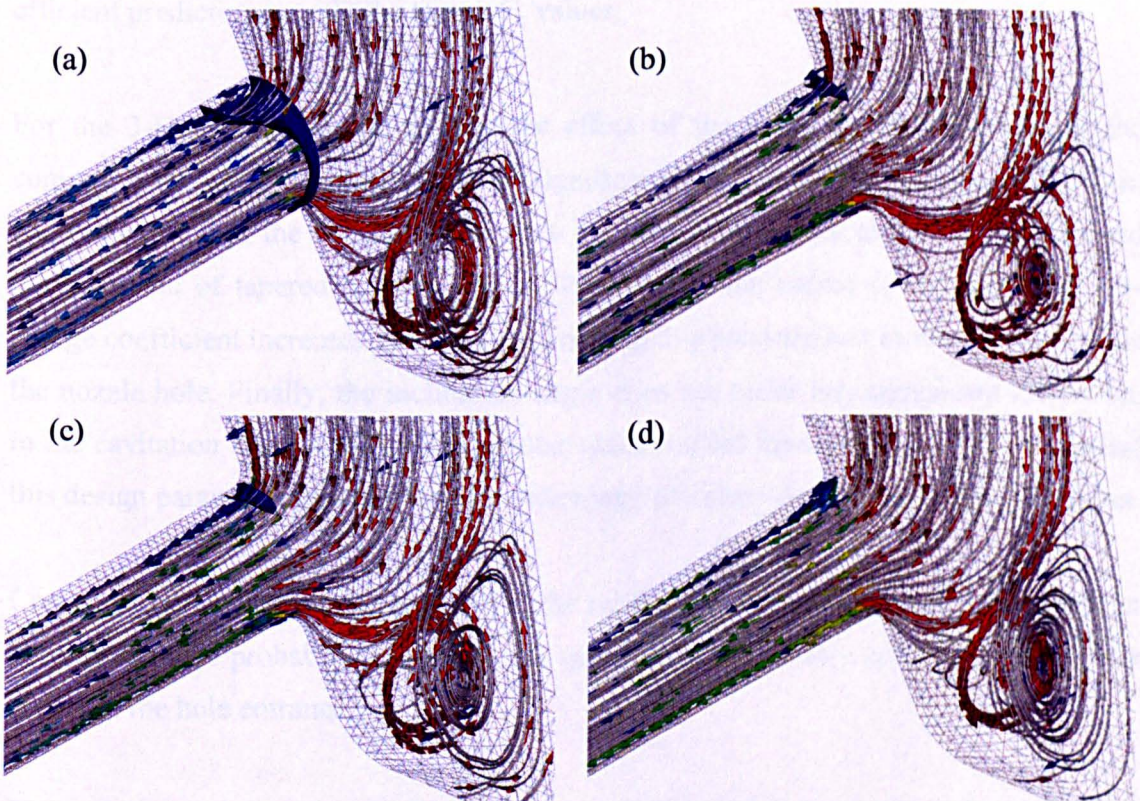


Figure 6.25: Initial (left) and optimised pressure contour of the mini-sac nozzle with constraint conicity (middle) and free parameters (right) needle lift after convergence of the cost function



On the same Figure 6.25 two views of the initial and predicted geometries of the two tests conducted are presented. As it has already been mentioned, the red contour corresponds to the positive pressure while the negative is represented with the other colours. Clearly the constraint geometry has still a small negative pressure area while in the other case the result is non-cavitating. It is also obvious that the constraint geometry predicted is also more curved than the unconstraint one.

The initial, predicted and two in-between iterations of the unconstraint case are presented in Figure 6.26 where the streamlines and the iso-surfaces of the negative pressure area are depicted. In Figure 6.26(a), (b) and (c) negative pressure still appear to the hole inlet where in the converged geometry in Figure 6.26(d) they are eliminated. The noticeable thing in this figure is that the recirculation in the sac becomes more intense while the iterations progress.



**Figure 6.26: Streamlines and iso-surface of negative pressure of the (a) initial, (b) fourth cycle, (c) eighth cycle and (d) predicted geometries for the mini-sac nozzle case**

## 6.7 SUMMARY

In this chapter four different geometries were used for the application of the developed model; the 2-D axisymmetric single-hole nozzle and three 3-D diesel nozzles. One of the 3-D nozzles was constructed for the model testing of this thesis while the other two are production nozzles, available to the group as part of recent and ongoing group research on cavitation. All these geometries were optimised with respect to the discharge coefficient and different tests have been conducted for each of them. From the investigation of this chapter several conclusions were drawn.

For the 2-D axisymmetric geometry, using just the conicity the target  $\bar{C}_d$  cannot be achieved in all cases due to the inefficiency of the parameterisation scheme. Different behaviour is noticed in the 3-D geometries where the conicity is modified using the exit instead of the inlet diameter. Using just the radius of curvature the model was more efficient predicting exactly the target  $\bar{C}_d$  values.

For the 3-D geometries, concerning the effect of the design parameters clearly the conicity is the one that causes the most significant increase in the discharge coefficient and elimination of the volume of negative pressure. This is due to the aforementioned characteristic of tapered nozzles. By the variation of the radius of curvature the discharge coefficient increases while the area of negative pressure just moves further inside the nozzle hole. Finally, the inclination angle does not make any significant difference in the cavitation intensity for the particular cases studied here. Nevertheless, in general this design parameter cannot vary in a wide range of values due to engine specifications.

Concerning the effect of the needle lift, the cavitation inception is easier eliminated in the low lift case probably because in this case cavitation usually appears in the needle seat than the hole entrance.

The conclusion that can be drawn from constraining certain parameters is that the model is varying the rest of the design parameters in order to converge but the predicted geometry may not have the expected performance since the search space for the optimisation is narrowed.



A general conclusion is that by using a combination of the design parameters instead of each one separately, the resulting geometry can achieve largest  $C_d$  values and the cavitation inception can, in some cases, be completely eliminated. Each design parameter in the combined case needed less modification to predict a larger  $C_d$  value than the extreme cases of each parameter by itself. This is a very convenient result since usually the manufacturers are seeking for improvement in the performance resulting by the least geometry modifications. The model also converges much faster in a multi-dimensional search space.

## *Chapter 7*

# *CONCLUSIONS & RECOMMENDATIONS FOR FUTURE WORK*

### **7.1 OVERVIEW**

The objective of the research documented in this thesis was the development of an automated mathematical methodology for designing nozzle geometries with specified flow characteristics. More specifically, the goal was to design automotive diesel injector nozzles able to produce controlled (or reduced) cavitation. The developed method is based on the adjoint formulation of the flow equations; in the case of this thesis the steady incompressible Navier-Stokes equations. The adjoint formulation is carried out by enforcing the flow equation in the form of constraints into the cost function. In this way gradients for each design parameter are calculated and driven to zero in order to direct the design parameters into an optimum geometry. The adjoint method has been chosen for the optimisation due to its accuracy and fast convergence. Furthermore, in nozzle re-design cases the required parameters describing the geometry and affecting cavitation may be numerous. The adjoint method provides the gradients of the cost function in a way that the computational effort required for this calculation is independent of the number of the design variables.

In order to setup the optimisation problem using the aforementioned method some tools needed to be developed. The cost functions were defined in such a way that by driving them to zero, geometries satisfying different flow characteristics were provided. The cost function is the means to quantify the optimum solution; as such it is really important to define it appropriately for each problem. A parameterisation scheme suitable for the design of automotive injector nozzles was developed and implemented in the code followed by a grid modification tool. Finally, an optimisation method completed the iterative optimisation process by driving the calculated gradients to zero and providing

the design parameters for the modification of the geometry towards the optimum solution.

To investigate the performance of the proposed methodology, the effect of various numerical parameters of the flow and adjoint equations, as well as parameters related to the optimisation method was explored. The validation of the model was carried out by means of inverse design with respect to a prescribed pressure distribution of a 2-D axisymmetric and a 3-D multi-hole VCO nozzle. For the purpose of optimising the geometries a cost function aiming to the maximisation of the discharge coefficient was used. At the same time, maximisation of the discharge coefficient led to geometries which have controlled or eliminated cavitation inception in the hole entrance. This is identified in the steady state mode by reduction of the volume of negative pressure appearing in the hole entrance. Several test cases, both 2-D and 3-D, were investigated. Parametric studies of the effect of the different design parameters on the optimum geometry are presented. The main conclusions drawn from the validation process and the parametric studies are presented in the following section, together with the conclusions from the application of the method for cavitation control. Subsequently, the recommended future work will be discussed in detail.

## 7.2 CONCLUSIONS

Before the model was applied for the optimisation of nozzle geometries with respect to cavitation control, various numerical and optimisation parametric investigations as well as validation cases were undertaken. Those cases consisted of the matching of a prescribed pressure distribution at the hole inlet of a 2-D axisymmetric nozzle for the parametric studies and a typical 3-D VCO nozzle for the validation. The main conclusions drawn from these studies are presented:

- Concerning the numerical parameters the effect of the grid and the effect of the number of iterations until convergence in the flow and adjoint solution were investigated. On the effect of the grid the conclusion can be drawn that the method works for various spatial discretisations. Moreover, the medium grid was selected due to the less computational time used without loss in accuracy of the predicted results. For the same reason in the parametric study of the depth of

convergence in the flow and adjoint solution the continuation scheme with the least possible iterations was proven to be the most efficient choice.

- Concerning the optimisation parameters, the effect of the steepest descent step size and the choices of target pressure distribution and initial geometry were investigated. When the step size is un-dimensionalised, increasing its value, within the order of magnitude of the design variables, leads to a faster convergence rate; up to a limit where the method will fail to converge. The case where the effect of different initial geometries was investigated showed robustness of the method. In all four tests the target pressure distribution was reproduced while the predicted geometries matched the target one. Finally, three different target distributions were considered and the model was setup to reproduce the target geometries. The conclusion was that the target pressure distributions can be reproduced and the predicted geometries approximated the target ones. Nevertheless, the results can be affected by the quality of the target geometry, which plays an important part in the convergence of the method.
- Another validation case was conducted using a typical 3-D VCO multi-hole Diesel nozzle and trying to match a pre-described pressure distribution at the hole inlet. Four design parameters were used in order to reproduce the in-between and resulting geometries of the design process. The optimisation method converges to the predescribed pressure distribution in less than 20 cycles while the predicted geometry closely approximates the target one.

The application of the model to cavitation control was conducted by the maximisation of the discharge coefficient and the subsequent reduction of the volume of negative pressure in the hole entrance. From this application several conclusions were drawn:

- The effect of each different design parameter on the results of the optimisation procedure was investigated and the conclusions are stated below. Clearly the conicity is the parameter that causes the most significant increase in the discharge coefficient. This stands true for tapered nozzles where a more gradual pressure drop happens along the length of the hole resulting to overall higher  $C_d$ . Another important effect of the conicity is that cavitation free nozzles can be predicted. On the other hand, the variation of the radius of curvature increases

the discharge coefficient while moving the area of negative pressure further inside the nozzle hole. In the curved case, the volume of negative pressure does not decrease significantly, but its intensity also decreases in absolute terms. Finally, the inclination angle when acting as the unique design variable does not make any significant difference in the cavitation intensity for the particular cases tested here. This is not a general conclusion and it is believed to occur due to the shape of the nozzle's sac and the position of the hole. Nevertheless, this design parameter is usually restricted within a small range of variation, as specified by the engine design.

- Concerning the effect of the needle lift, another test was undertaken. The conclusion drawn was that the cavitation inception is easier to eliminate in the low lift case. The likely cause of this effect is that at low lift conditions usually significant pressure drop occurs at the needle seat area, which results to a reduced pressure just upstream of the injection hole, and thus, reduced possibility for cavitation inception.
- A case was conducted where the design parameter producing conicity in the geometry was constrained. The conclusion that can be drawn from constraining certain parameters is that the model varies the rest of the design parameters more in order to converge. Nevertheless, the predicted geometry may not have the expected performance since the search space for the optimised geometry is narrowed.
- The following general conclusion can be drawn from the application of the model. By using a combination of the design parameters instead of each one individually, the resulting geometry can achieve higher  $C_d$  values and the cavitation inception can, in some cases, be completely eliminated. Each design parameter in the combined case required less modification to predict a larger  $C_d$  value than the extreme cases of each individual parameter. This is an important result since the manufacturers are seeking for improvement in the performance resulting by the fewer geometry modifications.

In order to summarise:



- The method can predict geometries with reduced or even eliminated geometrical cavitation as this is specified in the cost function. The method is fully automated and converges in 5 to 20 iterations according to the complexity of the specific case. The computational cost is reduced by the use of continuation in the solution of the flow and adjoint equations. Moreover, the method is robust and converges for different choices of initial geometries and target pressure distributions or target  $\bar{C}_d$ . Finally, the method is general and can be applied in other types of problem or geometries, for example airfoils.

### 7.3 RECOMMENDATIONS FOR FUTURE WORK

Design optimisation of internal flows is a recently initiated research area, which leaves many interesting possibilities for future work. In the remaining part of this chapter the most important recommendations for the extension and improvement of the developed optimisation model are presented:

- The developed methodology has the advantage of being cost independent of the number of parameters used in the problem, which is very helpful for 3-D applications. In these applications more complicated parameterisation schemes are required for describing the nozzle design. In this study the parameterisation of the geometry was limited to the shape of only the injection hole rather than the whole nozzle tip. With the optimisation facilities of the already developed method there are a lot of possibilities for further development by enhancing an automated parameterisation and grid generation tool. Such a commercial tool will give the freedom required to modify additional elements of the design such as the shape of the needle, the hole position or even the sac geometry.
- The model is currently restricted to single-phase and steady-state (without cavitation) calculations, for fixed needle lift position and constant inlet and exit pressure boundary conditions. Implementation of the optimiser in the cavitating flow would give the model the ability to produce geometries with more specified cavitation effects. This could also be a step forward in the design of allow improved designs of emerging multi-hole Diesel nozzles, capable of producing well atomised tailored sprays.

- Some other cost functions could be tested in order to solve other optimisation problems or to approach the problem of controlling cavitation from another point of view. For example, the volume of negative pressure can be included in the function that costs the improvement of the geometry. Other cost function choices could include the increase of the pressure values where they drop below zero or the maintenance of the flow rate between the inlet and exit.
- Improvements in the rate of convergence can be achieved by enhancing more sophisticated descent methods than steepest descent such as conjugate gradients and/or using adaptive step. Further improvements could be implemented in the “handling” of the geometry constraints to avoid non-feasible resulting geometries or geometries that do not satisfy manufacturing standards.

In the long run, computational simulation should become the principal tool for the engine design process because of the flexibility it provides for the rapid and comparatively inexpensive evaluation of alternative designs. Furthermore, it can be integrated in a numerical design environment providing for both multi-disciplinary analysis and multi-disciplinary optimisation. Developing an automated tool, which by the push of a button can actually produce nozzles leading to the desired engine performance, is definitely a very challenging task. By no means do we believe that the outcome of this thesis is capable of producing such results, but the author likes to think of this study as the first step towards this direction.

## *REFERENCES*

1. Trevena, D.H., *Cavitation and Tension in Liquids*. Bristol and Philadelphia: Adam Hilger. 1987.
2. Brennen, C.E., *Cavitation and Bubble Dynamics*. New York; Oxford: Oxford University Press. 1995.
3. Knapp, R.T., J.W. Daily, and F.G. Hammitt, *Cavitation*. New York; London: McGraw-Hill. 1970.
4. Pironneau, O., *On Optimum Design in Fluid Mechanics*. J. Fluid Mech, **64**: pp. 97-110, 1974.
5. Pironneau, O., *Optimal Shape Design for Elliptic Systems*. New York: Springer Verlag. 1984.
6. Jameson, A., *Aerodynamic Design via Control Theory*, in *ICASE Rep. no. 88-64*, 1988.
7. Anderson, W.K. and V. Venkatakrishnan, *Aerodynamic Design Optimisations on Unstructured Grids with a Continuous Adjoint Formulation*. AIAA Paper 97-0643, 1997.
8. Item, C.C. and O. Baysal, *Wing-section Optimisation for Super-sonic Viscous Flow*. ASME CFD for Design and Optimisation, **232**: pp. 29-36, 1995.
9. Jameson, A. and J.C. Vassberg, *Computational Fluid Dynamics for Aerodynamic Design: Its Current and Future Impact*, in *39th AIAA Aerospace Sciences Meeting and Exhibit*: Reno, 2001.
10. Labrujere, T.E. and J.V.D. Vooren, *Calculus of variations applied to 2d multi-point airfoil design*, in *The First European Computational Fluid Dynamics Conference*, 1992.
11. MacCormack, R.W., *Current Status of Numerical Solutions of the Navier-Stokes Equations*, in *AIAA Paper 85-0032, AIAA 23rd Aerospace Sciences Meeting*: Reno, 1985.
12. Ni, R.H., *A Multiple Grid Scheme for Solving the Euler Equations*. AIAA Journal, **20**: pp. 1565-1571, 1982.

13. Pulliam, T.H. and J.L. Steger, *Recent Improvements in Efficiency, Accuracy and Convergence for Implicit Approximate Factorization Algorithms*, in *AIAA Paper 85-0360, AIAA 23rd Aerospace Sciences Meeting*: Reno, 1985.
14. Murman, E.M. and J.D. Cole, *Calculation of Plane Steady Transonic Flows*. *AIAA Journal*, 9: pp. 114-121, 1971.
15. Jameson, A., *Iterative Solution of Transonic Flows Over Airfoils and Wings, Including Flows at Mach 1*. *Comm. Pure. Appl. Math*, 27: pp. 283-309, 1974.
16. Jameson, A., *Successes and Challenges in Computational Aerodynamics*, in *AIAA Paper 87-1184-CP, 8th Computational Fluid Dynamics Conference*: Hawaii, 1987.
17. Bristeau, M.O., O. Pironneau, R. Glowinski, J. Periaux, P. Perrier, and G. Poirier. *On the Numerical Solution of Nonlinear Problems in Fluid Dynamics by Least Squares and Finite Element Methods (II): Application to Transonic Flow Simulations*. in *Proceedings of 3rd International Conference on Finite Elements in Nonlinear Mechanics, FENOMECH 84*. Stuttgart. 1984.
18. Giannadakis, E., *Modelling of Cavitation in Automotive Fuel Injector Nozzles*. PhD Thesis, Imperial College, University of London, 2005.
19. Jameson, A. and D.A. Caughey. *A Finite Volume Method for Transonic Potential Flow Calculations*. in *Proceedings of AIAA 3rd Computational Fluid Dynamics Conference*. Albuquerque. 1977.
20. Jameson, A., W. Schmidt, and E. Turkel, *Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes*, in *AIAA Paper 81-1259, AIAA 14th Fluid Dynamics and Plasma Dynamics Conference*: Palo, Alto, 1981.
21. Jameson, A., T.J. Baker, and N.P. Weatherill, *Calculation of Inviscid Transonic Flow Over a Complete Aircraft*, in *AIAA Paper 86-0103, AIAA 24th Aerospace Sciences Meeting*: Reno, 1986.
22. Lighthill, M.J., *A New Method of Two Dimensional Aerodynamic Design*, in *R & M 1111, Aeronautical Research Council*, 1945.
23. McFadden, G.B., *An Artificial Viscosity Method for the Design of Supercritical Airfoils*, in *University Report C00-3077-158*: New York, 1979.
24. Taverna, F., *Advanced Airfoil Design for General Aviation Propellers*. *AIAA Paper 83-1791*, 1983.
25. Garabedian, P.R. and G. McFadden. *Computational Fluid Dynamics of Airfoils and Wings*. in *Proceedings of Symposium on Transonic, Shock, and Multidimensional Flows*. Madison: Academic Press, New York. 1981.

- 
26. Tranen, J.L., *A Rapid Computer Aided Transonic Airfoil Design Method*. AIAA Paper 74-5017, 1974.
  27. Volpe, G. and R.E. Melnik, *The Design of Transonic Aerofoils by a Well Posed Inverse Method*. Int. J. Numerical Methods in Engineering, **22**: pp. 341-361, 1986.
  28. Henne, P.A., *An Inverse Transonic Wing Design Method*. AIAA Paper 80-0330, 1980.
  29. Garabedian, P.R. and D.G. Korn. *Numerical Design of Transonic Airfoils*. in *Proceedings of SYNPADE 1970*: Academic Press, New York. 1970.
  30. Giles, M., M. Drela, and W.T. Thompkins. *Newton Solution of Direct and Inverse Transonic Euler Equations*. in *AIAA Paper 85-1530, Proceedings of AIAA 7th Computational Fluid Dynamics Conference*. Cincinnati. 1985.
  31. Hicks, R.M., E.M. Murman, and G.N. Vanderplaats, *An Assessment of Airfoil Design by Numerical Optimisation*, in *NASA Technical Memorandum X-3092*, 1974.
  32. Hicks, R.M. and P.A. Henne, *Wing Design by Numerical Optimization*. AIAA Paper 79-0080, 1979.
  33. Lions, J.L., *Optimal Control of Systems Governed by Partial Differential Equations*. New York: Springer Verlag. 1971.
  34. Jameson, A. and L. Martinelli, *Aerodynamic Shape Optimization Techniques Based On Control Theory*, in *International Mathematical Summer Center*: Martina Franca, Italy, 1999.
  35. Nocedal, J. and S.J. Wright, *Numerical Optimization* Springer Series in Operations Research: Springer 1999.
  36. Floudas, C. and P. Pardalos, eds. *Recent Advances in Global Optimization*. Princeton University Press: Princeton, NJ. 1992.
  37. Giannakoglou, K.C., A.P. Giotis, and M.K. Karakasis, *Low-Cost Genetic Optimisation based on Inexact Pre-evaluations and the Sensitivity Analysis of Design Parameters*. Inverse Problems in Engineering, **9**: pp. 389-412, 2001.
  38. Forth, S.A., D. Standingford, and P. Dawson, *An Adjoint Solver for an Industrial CFD Code via Automatic Differentiation*, in *AD2004: 4th International Conference on Automatic Differentiation*: University of Chicago, Gleacher Center, Chicago, USA 2004.



39. Hu, H., *Sensitivity Derivative Versions of the FLOMGEM CFD Code via Automatic Differentiation*, in *AIAA Report 97-1851*, 1997.
40. Pandya, M.J. and O. Baysal, *Gradient-based aerodynamic shape optimisation using alternating direction implicit method*. *Journal of Aircraft*, **34**(3): pp. 346-352, 1997.
41. Tortorelli, D.A., M.M. Tillier, and J.A. Dantzig, *Optimal Design of Nonlinear Parabolic System. Part I: Fixed Spatial Domain with Applications to Process Optimisation*. *Computer Methods in Applied Mechanics and Engineering*, **113**: pp. 141-155, 1994.
42. Wang, Z.X., D.A. Tortorelli, and J.A. Dantzig, *Sensitivity Analysis and Optimisation of Coupled Thermal and Flow Problems with Application to Contraction Design*. *Internal Journal for Numerical Methods for Fluids*, **23**: pp. 991-1020, 1996.
43. Anderson, W.K. and D.L. Bonhaus, *Aerodynamic Design on Unstructured Grids or Turbulent Flows*. NASA Technical Memorandum No. 112876, 1997.
44. Beux, F. and A. Dervieux, *Exact-gradient Shape Optimisation of a 2-D Euler Flow*. *Finite Elements in Analysis and Design*, **12**: pp. 281-302, 1992.
45. Pandya, M.J. and O. Baysal, *Three-Dimensional Viscous ADI Algorithm and Strategies for Shape Optimisation*, in *AIAA Report 97-1853*, 1997.
46. Reuther, J. and A. Jameson, *Aerodynamic Shape Optimisation of Wing and Wing-body Configurations Using Control Theory*, in *AIAA Report 95-0123*, 1995.
47. Alonso, J., A. Jameson, L. Martinelli, J. Reuther, and J. Vassberg, *An efficient Multi-block Method for Aerodynamic Analysis and Design on Distributed Memory Systems*. *AIAA Paper 97-1893*, 1997.
48. Rechenberg, I., *Evolution strategy: Optimisation of technical systems by means of biological evolution*, ed. Fromman-Holzboog. Stuttgart, Germany. 1973.
49. Chevalier, M., *Adjoint Based Control and Optimization of Aerodynamic Flows*, in *Department of Mechanics*. Royal Institute of Technology: Stockholm, Sweden, 2002.
50. Cauchy, A., *Methode generale pour la resolution des systemes d'equations simultanees*. *Comp. Rend. Acad. Sci. Paris*, **25**: pp. 536-538, 1847.
51. Fiacco, A.V. and G.P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, in *SIAM*: Philadelphia, 1990.

52. Arian, E., *A Preconditioning Method for Shape Optimization Governed by the Euler Equations*, in ICASE Report No. 98-14, 1998.
53. Arian, E. and S. Ta'asan, *Analysis of the Hessian for Aerodynamic Optimisation: Inviscid Flow*, in ICASE Report No. 96-28, 1996.
54. Reuther, J., J. Alonso, M.J. Rimlinger, and A. Jameson, *Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on distributed memory parallel computers*. Computers & Fluids, **28**: pp. 675-700, 1999.
55. Shewchuk, J.R., *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*, 1994.
56. Giles, M.B., *Aerospace design: a complex task*, in Report NA-97/07, VKI Lecture Series Course on Inverse Design, 1997.
57. Newman, J.C., A.C. Taylor, R.W. Barnwell, P.A. Newman, and G.J.-W. Hou, *Overview of sensitivity analysis and shape optimisation for complex aerodynamic configurations*. Journal of Aircraft, **36**: pp. 87-96, 1999.
58. Tortorelli, D.A. and P. Michaleris, *Design Sensitivity Analysis: Overview and Review*. Inverse Problems in Engineering, **1**: pp. 71-105, 1994.
59. Baysal, O. and M.E. Eleshaky, *Aerodynamic Design Optimisation Using Sensitivity Analysis and Computational Fluid Dynamics*. AIAA Journal, **30**(3), 1992.
60. Borggaard, J. and J. Burns, *A Sensitivity Equation Approach to Shape Optimisation in Fluid Flows*, in ICASE Report No. 94-8, 1994.
61. Burgreen, G.W. and O. Baysal, *Three-Dimensional aerodynamic shape optimisation of wings using discrete sensitivity analysis*. AIAA Journal, **34**: pp. 1761-1770 1996.
62. Burkardt, J. and M. Gunzburger, *Sensitivity Discrepancy for Geometric Parameters*. ASME CFD for Design and Optimisation, **232**(9-15), 1995.
63. Hou, G.J.-W., V. Maroju, A.C. Taylor, V.M. Korivi, and P.A. Newman, *Transonic Turbulent Airfoil Design Optimisation with Automatic Differentiation in Incremental Forms*. AIAA Paper 95-1692, 1995.
64. Eleshaky, M.E. and O. Baysal, *Airfoil Shape Optimisation Using Sensitivity Analysis on Viscous Flow Equations*. Journal of Fluids Engineering, **115**: pp. 75-84, 1993.

- 
65. LeGresley, P.A. and J.J. Alonso, *Investigation of Non-Linear Projection for POD Based Reduced Order Models for Aerodynamics*, in *39th AIAA Aerospace Sciences Meeting and Exhibit*: Reno, 2001.
  66. Korivi, V.M., A.C. Taylor, P.A. Newman, G.W. Hou, and H.E. Jones, *An Approximately Factored Incremental Strategy for Calculating Discrete Aerodynamic Sensitivity Derivatives*. *Journal of Computational Physics* **113**: pp. 336-346, 1994.
  67. Carle, A., M. Fagan, and L.L. Green, *Preliminary results from the application of automated codegeneration to CFL3D*, in *AIAA 98-4807*, 1998.
  68. Faure, C., *Splitting of algebraic expressions for automatic differentiation*, in *Proceedings of the second SIAM Int. Workshop on Computational Differentiation*, 1996.
  69. Giering, R. and T. Kaminski, *Recipes for Adjoint Code Construction*. *ACM trans. Math. Software*, **24**(4): pp. 437-474, 1998.
  70. Borggaard, J. and J. Burns, *A PDE Sensitivity Equation Method for Optimal Aerodynamic Design*. *Journal of Computational Physics*, **136**: pp. 366-384, 1997.
  71. Hazra, S.B., *An Efficient Method for Aerodynamic Shape Optimization*, in *AIAA 2004-4628, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*: Albany, New York, 2004.
  72. Martin, T.J. and G.S. Dulikravich, *An implicit and explicit BEM sensitivity approach for thermo-structural optimization*. *Engineering Analysis with Boundary Elements*, **28**: pp. 257-265, 2004.
  73. Balagangadhar, D. and S. Roy, *Design sensitivity analysis and optimization of steady fluid-thermal systems*. *Computer Methods in Applied Mechanics and Engineering*, **190**: pp. 5465-5479, 2001.
  74. Kammerer, S., J.F. Mayer, H. Stetter, M. Paffrath, U. Wever, and A.R. Jung, *Development of a Three-Dimensional Geometry Optimization Method for Turbomachinery Applications*. *International Journal of Rotating Machinery*, **10**(5): pp. 373-385, 2004.
  75. Shankaran, S., T. Doyle, M. Gerritsen, G. Iaccarino, and A. Jameson, *Improving the Design of Sails Using CFD and Optimization Algorithms*, in *High Performance Yacht Design Conference*: Auckland, 2002.
  76. Le Pape, A. and P. Beaumier, *Numerical optimization of helicopter rotor aerodynamic performance in hover*. *Aerospace Science and Technology*, **9**: pp. 191-201, 2005.

- 
77. Svenningsen, K.H., J.I. Madsen, N.H. Hassing, and H.G. Pauker, *Optimization of flow geometries applying quasianalytical sensitivity analysis*. Applied Mathematical Modelling **20**(3): pp. 214-224, 1996.
  78. Lund, E., H. Moller, and L.A. Jakobsen, *Shape Design Optimization of Stationary Fluid-Structure Interaction Problems With Large Displacements and Turbulence*, in *Proc. 4th World Congress on Structural and Multidisciplinary Optimization* Dalian, China, 2001.
  79. Li, W., S. Krist, and R. Campbell, *Transonic Airfoil Shape Optimization in Preliminary Design Environment*, in *10<sup>th</sup> AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference* 2004.
  80. Besnard, E. and A. Schmitz, *Two-Dimensional Aircraft High Lift System Design and Optimization*, in *AIAA Paper 98-0123*, 1998.
  81. Fogel, L.J., A.J. Owens, and M.J. Walsh, *Artificial Intelligence through Simulated Evolution*. New York, NY: John Wiley & Sons. 1996.
  82. Schwefel, H.P., *Numerical Optimisation of Computer Models*. Chichester, UK: John Wiley & Sons. 1981.
  83. Holland, J.H., *Adaptation in natural artificial systems*, in *University of Michigan Press*, M. Ann Arbor, Editor, 1975.
  84. Goldberg, D.E., *Genetic Algorithms in Search, Optimisation and Machine Learning*, ed. Addison-Wesley: Reading, MA. 1989.
  85. Mitchell, M., *An Introduction to Genetic Algorithms*, M. Press, Editor: Cambridge, MA, 1996.
  86. Muyl, F., L. Dumas, and V. Herbert, *Hybrid method for aerodynamic shape optimization in automotive industry*. Computers & Fluids, **33**: pp. 849-858, 2004.
  87. Hamalainen, J.P., R.A.E. Makinen, P. Tarvainen, and J. Toivanen, *Evolutionary Shape Optimization in CFD with Industrial Applications*, in *European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS 2000*: Barcelona, Spain, 2000.
  88. Han, S.-Y. and J.-S. Maeng, *Shape optimization of cut-off in a multi-blade fan/scroll system using neural network*. International Journal of Heat and Mass Transfer, **46**: pp. 2833–2839, 2003.
  89. Dornberger, R., P. Stoll, D. Buche, and A. Neu, *Multidisciplinary Turbomachinery Blade Design Optimization*, in *AIAA Paper 2000-0838*, 2000.

90. Bäck, T., *Evolutionary Algorithms in Theory and Practice: Evolutionary Strategies, Evolutionary Programming, Genetic Algorithms*. New York: Oxford University Press. 1996.
91. Fogel, D.B., *An introduction to simulated evolutionary optimisation*. IEEE Transactions on Neural Networks, **5**(1): pp. 3-14, 1994.
92. Spears, W.M., K.A. De Jong, T. Bäck, D.B. Fogel, and H. De Garis. *An overview of evolutionary computation*. in *Proceedings of the European Conference on Machine Learning (ECML-93)*. Berlin, Germany: Springer-Verlag. 1993.
93. Michalewicz, Z. and M. Michalewicz. *Evolutionary computation techniques and their applications*. in *Proceedings of the IEEE International Conference on Intelligent Processing Systems*. Piscataway, NJ: IEEE Publications. 1997.
94. Calegari, P., G. Coray, A. Hertz, D. Kobler, and P. Kuonen, *A taxonomy of evolutionary algorithms in combinatorial optimisation*. Journal of Heuristics, **5**: pp. 145-158, 1999.
95. Gauger, N. and J. Brezillon, *The Continous Adjoint Approach in Aerodynamic Shape Optimization*, in *MEGAFLOW - Numerical Flow Simulation for Aircraft Design*. Springer Verlag, 2005.
96. Giles, M.B., M.C. Duta, J.-D. Muller, and N.A. Pierce, *Algorithm developments for discrete adjoint methods*. AIAA Journal, **41**(2), 2003.
97. Jameson, A., *Optimum Aerodynamic Design Using CFD and Control Theory*, in *AIAA Paper 95-1729, AIAA 12th Computational Fluid Dynamics Conference*: San Diego, 1995.
98. Nadarajah, S. and A. Jameson, *A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimisation*, in *AIAA paper 2000-0667. 38th. Aerospace Sciences Meeting and Exhibit*: Reno, NV, 2000.
99. *Automatic Differentiation* Joint effort between Aachen and Argonne - <http://www.autodiff.org/>.
100. Giles, M.B., *On the use of Runge-Kutta time-marching and multigrid for the solution of steady adjoint equations*, in *Report NA-00/10*. Oxford University Computing Laboratory, 2000.
101. Giles, M.B., *On the iterative solution of adjoint equations*, in *Automatic Differentiation: From Simulation to Optimisation*, C.F. G. Corliss, A. Griewank, L. Hascoet and U. Neumann,, Editor. Springer-Verlag. pp. 145-152. 2001.
102. Duivesteijn, G.F., H. Bijl, B. Koren, and E.H. van Brummelen, *On the adjoint solution of the quasi-1D Euler equations: the effect of boundary conditions and*



- the numerical flux function*, in *MAS-E0533 Modelling, Analysis and Simulation*. Centrum voor Wiskunde en Informatica: Amsterdam, 2005.
103. Hiernaux, S. and J.-A. Essers, *A continuous adjoint approach to solve bidimensional shape optimization problems for external flows*, in *European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS 2000*: Barcelona, Spain, 2000.
  104. Monge, F. and B. Tobio, *Aerodynamic Design and Optimization by Means of Control Theory in COMPUTATIONAL MECHANICS New Trends and Applications CIMNE*: Barcelona, Spain, 1988.
  105. Ochiai, T. and M. Kawahara, *Shape optimization Problem for Incompressible Viscous Flow Based on Optimal Control Theory*. in *Proceedings of the eighth international conference on the application of artificial intelligence to civil and structural engineering computing*. Stirling, Scotland. 2001.
  106. Li, W. and L. Padula, *Robust Airfoil Optimization in High Resolution Design Space*, in *ICASE*. NASA Langley Research Center: Hampton, Virginia, 2002.
  107. Muller, J.-D. and P. Cusdin, *On the Performance of Discrete Adjoint CFD Codes using Automatic Differentiation*. *International Journal for Numerical Methods in Fluids*, 00: pp. 1-6, 2000.
  108. Kuruvila, G., S. Ta'asan, and M.D. Salas, *Airfoil Design and Optimisation by the One-Shot Method*, in *AIAA Report 95-0478*, 1995.
  109. Nielsen, E.J., W.K. Anderson, and D.K. Kaushik. *Implementation of a Parallel Framework for Aerodynamic Design Optimization on Unstructured Meshes*. in *Proceedings of Parallel CFD 99* 1999.
  110. Brezillon, J. and R. Dwight, *Discrete Adjoint of the Navier-Stokes Equations for Aerodynamic Shape Optimization*, in *EUROGEN 2005 - Sixth Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*: Munich, Germany, 2005.
  111. Nielsen, E.J. and W.L. Kleb, *Efficient Construction of Discrete Adjoint Operators on Unstructured Grids Using Complex Variables*. *AIAA Journal*, 44(4): pp. 827, 2006.
  112. Nemec, M., M.J. Aftosmis, S.M. Murman, and T. Pulliam, *Adjoint Formulation for an Embedded-Boundary Cartesian Method*, in *43rd AIAA Aerospace Sciences Meeting*: Reno, NV, 2005.
  113. Giles, M.B. and N.A. Pierce, *An introduction to the adjoint approach to design*. *Flow, Turbulence and Combustion*, 65(3-4): pp. 393-415, 2000.

114. Giles, M.B. and N.A. Pierce, *Adjoint Equations in CFD: Duality, Boundary Conditions, and Solution Behaviour*, in *AIAA Report 97-1850*, 1997.
115. Jameson, A., N.A. Pierce, and L. Martinelli, *Optimum Aerodynamic Design using the Navier-Stokes Equations*, in *AIAA Report 97-0101*, 1997.
116. Arian, E. and S. Ta'asan, *Admitting the Inadmissible: Adjoint Formulation for Incomplete Cost Functionals in Aerodynamic Optimization*, in *ICASE Report No. 97-69*, 1997.
117. Soemarwoto, B.I., *The Variational Method for Aerodynamic Optimisation Using the Navier-Stokes Equations*, in *ICASE Report No. 97-71*, 1997.
118. Giles, M.B. and N.A. Pierce, *On the properties of solutions of the adjoint Euler equations*, in *6th ICFD Conference on Numerical Methods for Fluid Dynamics*: Oxford, UK, 1998.
119. Giles, M.B. and N.A. Pierce, *Analytic adjoint solutions for the quasi-one-dimensional Euler equations*. *Journal of Fluid Mechanics*, **426**: pp. 327-345, 2001.
120. Iollo, A. and M.D. Salas, *Optimum Transonic Airfoils based on the Euler Equations*, in *ICASE Rep.no. 96-76*, 1996.
121. Giles, M.B., *Analysis of the Accuracy of Shock-Capturing in the Steady Quasi-ID Euler Equations*. *Computational Fluid Dynamics Journal*, **5**(2): pp. 247-258, 1996.
122. Giles, M.B., *Discrete adjoint approximations with shocks*, in *Hyperbolic Problems: Theory, Numerics, Applications*, T.H.a.E. Tadmor, Editor. Springer-Verlag. 2003.
123. Kim, S., J. Alonso, and A. Jameson, *A Gradient Accuracy Study for the Adjoint-Based Navier-Stokes Design Method*, in *AIAA 37th Aerospace Sciences Meeting and Exhibit*: Reno, NV, 1999.
124. Jameson, A. and J. Vassberg, *Studies of Alternative Numerical Optimization Methods Applied to the Brachistochrone Problem*. *Computational Fluid Dynamics Journal*, **9**(3): pp. 281-296, 2000.
125. Protas, B. and T.R. Bewley, *Regularization opportunities in the adjoint analysis of multiscale systems*, in *Advances in Turbulence IX Proceedings of the Ninth European Turbulence Conference*, CIMNE: Barcelona, Spain, 2002.
126. Jameson, A. and J. Reuther. *Control theory based airfoil design using the Euler equations*. in *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. Panama City Beach. 1994.

127. Jameson, A., *Essential Elements of Computational Algorithms for Aerodynamic Analysis Design*, in *ICASE Report No. 97-68*, 1997.
128. Jameson, A. and L. Martinelli, *Optimum Aerodynamic Design using the Navier-Stokes Equations*. Theoret. Comput. Fluid Dynamics, 10: pp. 213-237, 1998.
129. Reuther, J., A. Jameson, J. Farmer, L. Martinelli, and D. Saunders, *Aerodynamic Shape Optimisation of Complex Aircraft Configurations via an Adjoint Formulation*, in *AIAA Report 96-0094*, 1996.
130. Kim, S., K. Hosseini, K. Leoviriyakit, and A. Jameson, *Enhancement of Adjoint Design Methods via Optimization of Adjoint Parameters*, in *AIAA Paper 2005-0448, 43rd AIAA Aerospace Sciences Meeting and Exhibit*: Reno, NV, 2005.
131. Kim, S., J. Alonso, and A. Jameson, *Two-dimensional High-Lift Aerodynamic Optimization Using the Continuous Adjoint Method*, in *AIAA Paper 2000-4741, 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*: Long Beach, CA, 2000.
132. Kim, S., J. Alonso, and A. Jameson, *Design Optimization of High-Lift Configurations Using a Viscous Continuous Adjoint Method*, in *AIAA Paper 2002-0844, 40th AIAA Aerospace Sciences Meeting and Exhibit*: Reno, NV, 2002.
133. Nadarajah, S., A. Jameson, and J. Alonso, *An Adjoint Method for the Calculation of Non-Collocated Sensitivities in Supersonic Flow*, in *First MIT Conference on Computational Fluid Dynamics*: Cambridge, MA, 2001.
134. Nadarajah, S., A. Jameson, and J. Alonso, *An Adjoint Method for the Calculation of Remote Sensitivities in Supersonic Flow*, in *AIAA-2002-0261, 40th AIAA Aerospace Sciences Meeting and Exhibit*: Reno, NV, 2002.
135. Ta'asan, S., *Pseudo-Time Methods for Constrained Optimisation Problems Governed by PDEs*, in *ICASE Report No. 95-32*, 1995.
136. Ta'asan, S., *Introduction to Shape Design and Control*, in *Inverse Problems in Optimisation Methods, Von Karman Institute Lecture Series 97-05*, 1997.
137. Iollo, A., G. Kuruvila, and S. Ta'asan, *Pseudo-time method for optimal shape design using the Euler equations*. AIAA Journal, 34: pp. 1807-1813 1996.
138. Iollo, A., M.D. Salas, and S. Ta'asan, *Shape Optimisation Governed by the Euler Equations Using an Adjoint Method*, in *ICASE Rep. no. 93-78*, 1993.
139. Giles, M.B., *Aerodynamic Design Optimisation for Complex Geometries using Unstructured Grids*, in *VKI Lecture Course on Inverse Design* 1997.

140. Giles, M.B., M.C. Duta, and J.-D. Muller, *Adjoint Code Developments Using the Exact Discrete Approach*, in *AIAA Paper 2001-2596, 15th Computational Fluid Dynamics Conference*: Anaheim, California, 2001.
141. Campobasso, M.S., M.C. Duta, and M.B. Giles, *Adjoint Methods for Turbomachinery Design*, in *ISABE-2001-1055*, 2001.
142. Duta, M.C., M.B. Giles, and M.S. Campobasso, *The Harmonic Adjoint Approach to Unsteady Turbomachinery Design* International Journal for Numerical Methods in Fluids, **40**(3-4): pp. 323-332, 2002.
143. Campobasso, M.S. and M.B. Gilles, *Computing Linear Harmonic Unsteady Flows in Turbomachines with Complex Iterative Solvers*, in *AIAA CFD Conference*, 2005.
144. Huan, J. and V. Modi, *Optimum Design of Minimum Drag Bodies in Incompressible Laminar Flow Using a Control Theory Approach*. Inverse Problems in Engineering, **1**: pp. 1-25, 1994.
145. Soemarwoto, B.I., *Multi-Point Aerodynamic Design by Optimisation*. Delft University of Technology: the Netherlands, 1996.
146. Soto, O. and R. Löhner, *CFD Shape Optimisation Using an Incomplete-Gradient Adjoint Formulation*. Int.J. Num. Meth. Eng, **51**(735-753), 2001.
147. Soto, O. and R. Löhner, *General Methodologies for Incompressible Flow Design Problems*. AIAA-01-1061 2001.
148. Soto, O., R. Löhner, and C. Yang, *A Stabilized Pseudo-Shell Approach for Surface Parametrization in CFD Design Problems*. Comm. Num. Meth. Eng, **18**(4): pp. 251-258, 2002.
149. Soto, O. and R. Löhner, *A Mixed Adjoint Formulation for Incompressible RANS Problems*. AIAA-02-0451 2002.
150. Löhner, R., O. Soto, and C. Yang, *An Adjoint-based design methodology for CFD Optimization problems*, in *AIAA Paper 03-0299*, 2003.
151. Soto, O. and R. Löhner, *On the Computation of Flow Sensitivities from Boundary Integrals*, in *42nd AIAA Meeting and Exhibit Reno*, 2004.
152. LeGresley, P.A. and J. Alonso, *Improving the Performance of Design Decomposition Methods with POD*, in *AIAA Paper 2004-4465, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*: Albany, NY, 2004.

153. Hazra, S.B. and V. Schulz, *Simultaneous Pseudo-Timestepping for Aerodynamic Shape Optimization Problems with State Constraints*. Bit Numerical Mathematics, 44(3): pp. 457-472, 2004.
154. Petropoulou, S., T. Pappou, D. Koubogiannis, and G. Freskos, *Multi-point Airfoil Design Using a Continuous Adjoint Method*, in *CEAS Aerospace Aerodynamics Research Conference 2002*: Cambridge, United Kingdom, 2002.
155. Reuther, J., A. Jameson, J.J. Alonso, M.J. Rimlinger, and D. Saunders, *Constrained Multi-point Aerodynamic Shape Optimisation Using an Adjoint Formulation and Parallel Computers*, in *AIAA Report 97-0103*, 1997.
156. Leoviriyakit, K. and A. Jameson, *Multi-point Wing Planform Optimization via Control Theory*, in *43rd AIAA Aerospace Sciences Meeting & Exhibit, AIAA Paper 2005-0450*: Reno, NV, 2005.
157. Wild, J. and J. Brezillon, *Application of multi-objective constraint optimization in aerodynamic high-lift design*, in *Proceedings of the Sixth Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*. TU München: München, BY (D), 2005.
158. Nemec, M. and M.J. Aftosmis, *Adjoint Algorithm for CAD-Based Shape Optimization Using a Cartesian Method*, in *AIAA Paper 2005-4987*, 2005.
159. Giannakoglou, K.C., T.I. Pappou, A.P. Giotis, and D.G. Koubogiannis, *A Parallel Inverse-Design Algorithm in Aeronautics Based on Genetic Algorithms and the Adjoint Method*, in *ECCOMAS2000*: Barcelona, Spain, 2000.
160. Nielsen, E.J. and M.A. Park, *Using an Adjoint Approach to Eliminate Mesh Sensitivities in Computational Design*. AIAA JOURNAL, 44(5): pp. 948, 2006.
161. Cowles, G. and L. Martinelli, *A control-theory based method for shape design in incompressible flow using RANS*, in *Proceedings of the Fluids 2000 Conference and Exhibit*: Denver, CO, 2000.
162. Tahara, Y., E. Paterson, F. Stern, and Y. Himeno. *CFD-based optimisation of naval/surface combatant*. in *Proceedings of the 23rd Symposium on Naval Hydrodynamics*. Val de Reuil, France. 2000.
163. Dreyer, J. and L. Martinelli, *Hydrodynamic shape optimisation of propulsor configurations using a continuous adjoint approach*, in *Proceedings of the 15th AIAA Computational Fluid Dynamics Conference*: Anaheim, CA, 2000.
164. Giles, M.B., M.G. Larson, J.M. Levenstam, and E. Suli, *Adaptive error control for finite element approximations of the lift and drag coefficients in viscous flow*, in *Report NA-97/06*. Oxford University Computing Laboratory, 1997.



165. Giles, M.B., *On adjoint equations for error analysis and optimal grid adaptation*, in *Automatic Differentiation: From Simulation to Optimisation*, D.A.C.a.M.M. Hafez, Editor. World Scientific. pp. 155-170. 1998.
166. Giles, M.B. and N.A. Pierce, *Adjoint recovery of superconvergent functionals from approximate solutions of partial differential equations*. SIAM Review, **42**(2): pp. 247-264, 2000.
167. Giles, M.B. and N.A. Pierce, *Superconvergent lift estimates through adjoint error analysis*, in *Special meeting to honour Prof. Phil Roe*: Arcachon, 1998.
168. Muller, J.-D. and M.B. Giles, *Solution Adaptive Mesh Refinement Using Adjoint Error Analysis*, in *AIAA Paper 2001-2550, 15th Computational Fluid Dynamics Conference*: Anaheim, California, 2001.
169. Joslin, R.D., M.D. Gunzburger, R.A. Nicolaides, G. Erlebacher, and M.Y. Hussaini, *A self-contained, Automated Methodology for Optimal Flow Control Validated for Transition Delay*, in *ICASE Report No. 96-64*, 1996.
170. Gilles, M.B. and P. Glasserman, *Smoking Adjoint: fast evaluation of Greeks in Monte Carlo calculations*. Oxford University Computing Laboratory, 2005.
171. Arcoumanis, C., M. Badami, H. Flora, and M. Gavaises, *Cavitation in Real-Size Multi-Hole Diesel Injector Nozzles*. SAE Paper 2000-01-1249, 2000.
172. Afzal, H., C. Arcoumanis, M. Gavaises, and N. Kampanis, *Internal Flow in Diesel Injector Nozzles: Modelling and Experiments*. IMechE Paper S492/S2/99, 1999.
173. Roth, H., M. Gavaises, and C. Arcoumanis, *Cavitation Initiation, Its Development and Link with Flow Turbulence in Diesel Injector Nozzles*. SAE Paper 2002-01-0214, 2002.
174. Chen, Y.L. and S.D. Heister, *Two-Phase Modeling of Cavitated Flows*. Computers & Fluids, **24**(7): pp. 799-809, 1995.
175. Marcer, R. and J.M. LeGouez, *Simulation of Unsteady Cavitating Flows in Diesel Injector with an Improved VOF Method*, in *Proc. ILASS-EUROPE*: Zurich, Switzerland, 2001.
176. Yuan, W., J. Sauer, and G.H. Schnerr, *Modelling and Computation of Unsteady Cavitation Flows in Injection Nozzles*, in *1st International Colloquium on Microhydrodynamics*: Paris, France, 2000.
177. Schmidt, D.P., C.J. Rutland, and M.L. Corradini, *A Numerical Study of Cavitating Flow Through Various Nozzle Shapes*. SAE Paper 971597, 1997.

- 
178. Schmidt, D.P., C.J. Rutland, M.L. Corradini, P. Roosen, and O. Genge, *Cavitation in Two-Dimensional Asymmetric Nozzles*, in *SAE 1999-01-0518, SAE International Congress and Exposition*: Detroit, Michigan, 1999.
179. Arcoumanis, C. and M. Gavaises. *Cavitation in Diesel Injectors: Modelling and Experiments*. in *Proc. ILASS-EUROPE*. Manchester, UK. 1998.
180. Giannadakis, E., M. Gavaises, H. Roth, and C. Arcoumanis, *Cavitation Modelling in Single-Hole Diesel Injector Based on Eulerian-Lagrangian Approach*, in *Proc. THIESEL International Conference on Thermo- and Fluid Dynamic Processes in Diesel Engines*: Valencia, Spain, 2004.
181. Matsumoto, Y., T. Kanbara, K. Sugiyama, and Y. Tamura. *Numerical Study of Cavitating Flow Structure on a Hydrofoil*. in *Proc. 4th KSME-JSME Fluids Engineering Conference*. Haeundae, Pusan, Korea. 1998.
182. Hsiao, C.-T., G.L. Chahine, and H.L. Liu, *Scaling effect on prediction of cavitation inception in a line vortex flow*. *Journal of Fluids Engineering-Transactions of the ASME*, **125**(1): pp. 53-60, 2003.
183. Choi, T.J., C.H. Amon, T.I.-P. Shih, and N. Trigui, *CFD Shape Optimisation Based on an Adjoint Variable Formulation of the Compressible Navier – Stokes Equations*, in *AIAA Paper 457*: Reno, Nevada 2000.
184. Lim, S. and H. Choi, *Optimal Shape Design of a Two-Dimensional Asymmetric Diffuser in Turbulent Flow*. *AIAA Journal*, **42**(6): pp. 1154-1169, 2004.
185. Kaminski, T., R. Giering, and C. Othmer, *Topological Design Based on Highly Efficient Adjoints Generated by Automatic Differentiation*. to appear in *ERCOFTAC 2006, Design an Optimisation: Methods and Applications*, 2006.
186. Trigui, N., V. Giaznov, H. Affes, and D. Smith, *CFD Based Shape Optimisation of IC Engine*. *Oil & Gas Science and Technology – Rev. IFP*, **54**(2), 1999.
187. Madsen, J.I., *Design optimization of internal flow devices*, in *Institute of Energy and Institute of Mechanical Engineering*. Aalborg University: Aalborg East, Denmark, 1998.
188. Brewer, W.H., J.C. Newman, G.W. Burgreen, and C.O.E. Burg, *A Design Method for Investigating Cavitation Delay*, in *8th Int. Conf. on Numerical Ship Hydrodynamics*: Busan, Korea, 2003.
189. Bergin, M., R.P. Hessel, and R.D. Reitz, *Optimization of a Large Diesel Engine via Spin Spray Combustion*. *SAE Paper 2005-01-0916*, 2005.
190. Munnannur, A., S.-C. Kong, and R.D. Reitz, *Performance Optimization of Diesel Engines with Variable Intake Valve Timing via Genetic Algorithms*. *SAE Paper 2005-01-0374*, 2005.

- 
191. Koubogiannis, D.G., L.C. Pousoulidis, D.V. Rovas, and K.C. Giannakoglou, *Solution of Flow Problems Using Unstructured Grids on Distributed Memory Platforms*. Computer Methods in Applied Mechanics and Engineering, **160**: pp. 89-100, 1998.
  192. *Aeroshape book*. Notes in Num. Fluid Mechanics, ed. V.e.a. Selmin: Springer. 2003, to appear.
  193. Ta'asan, S., *Theoretical Tools for Problem Setup*, in *Inverse Problems in Optimisation Methods, Von Karman Institute Lecture Series 97-05*, 1997.
  194. Gavaises, M. and A. Andriotis, *Cavitation Inside Multi-hole Injectors for Large Diesel Engines and its Effect on the Near-nozzle Spray Structure*. SAE Paper 06-01-1114, 2006.
  195. Olhoff, N. and J.E. Taylor, *On Structural Optimisation* Journal of Applied Mechanics, **50**: pp. 1139-1151, 1983.
  196. Jameson, A., *Computational Aerodynamics for Aircraft Design*. Science, **245**: pp. 361-371, 1989.
  197. Arian, E. and S. Ta'asan. *Smoothers for Optimization problems*. in *Proceedings of the Seventh Copper Mountain Conference on Multigrid Methods*. 1995.
  198. Thompson, J.F., Z.U.A. Warsi, and M. C.W., *Numerical grid generation – foundations and applications*. New York Elsevier. 1985.
  199. Arcilla, A.S., J. Hauser, P.R. Eiseman, and J.F. Thompson, *Numerical grid generation in computational fluid dynamics and related fields*. North Holland, Amsterdam. 1991.
  200. Launder, B.E. and D.B. Spalding, *The Numerical Computation of Turbulent Flows*. Computer Methods in Applied Mechanics and Engineering, **3**: pp. 269-289, 1974.
  201. Launder, B.E. and D.B. Spalding, *Lectures in Mathematical Models of Turbulence*. London: Academic Press. 1972.
  202. Vrahatis, M.N., G.S. Androulakis, J.N. Lambrinos, and G.D. Magoulas, *A class of gradient unconstrained minimization algorithms with adaptive stepsize*. Journal of Computational and Applied Mathematics, **114**: pp. 367-386, 2000.
  203. Hirsch, C., *Numerical Computation of Internal and External Flows, Vol. I&II*": Wiley. 1990.
  204. Ferziger, J.H. and M. Peric, *Computational Methods for Fluid Dynamics*. 2nd ed. Berlin Heidelberg: Springer-Verlag. 1996.

- 
205. Versteeg, H.K. and W. Malalasekera, *An Introduction to Computational Fluid Dynamics. The Finite Volume Method.*: Longman Scientific & Technical. 1995.
206. Jasak, H., *Error analysis and estimation for the finite volume method with applications to fluid flows*. PhD Thesis, Imperial College, University of London, 1996.
207. Jasak, H., H.G. Weller, and A.D. Gosman, *High resolution NVD differencing scheme for arbitrarily unstructured meshes*. International Journal for Numerical Methods in Fluids, **31**(2): pp. 431-449, 1999.
208. Ubbink, O., *Numerical prediction of two fluid systems with sharp interfaces*, in *Department of Mechanical Engineering, Imperial College of Science, Technology & Medicine*. University of London: London, 1997.
209. Hirsch, C., *Numerical Computation of Internal and External Flows*. Vol. I & II: John Wiley & Sons. 1991.
210. SLAP, *Sparse Linear Algebra Package*. Netlib Repository - <http://www.netlib.org/>.
211. Tong, C.H. and Q. Ye, *Analysis of the Finite Precision Bi-conjugate Gradient Algorithm for Nonsymmetric Linear Systems* Mathematics of Computation, **69**(232): pp. 1559-1575, 1999.
212. Caretto, L.S., A.D. Gosman, S.V. Patankar, and D.B. Spalding. *Two calculation procedures for steady, three-dimensional flows with recirculation*. in *Proc. Third International Conference on Numerical Methods in Fluid Dynamics*. Paris, France. 1972.
213. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*: Taylor & Francis Inc. 1980.
214. Rhie, C.M. and W.L. Chow, *A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation*. AIAA Journal, **21**: pp. 1525-1532, 1983.
215. Issa, R.I., *Solution of implicitly discretised fluid flow equations by operator-splitting*. Journal of Computational Physics, **62**: pp. 40-65, 1986.
216. Bosch. Robert Bosch GmbH (FV/SLE), Stuttgart, Germany.
217. König, G. and M. Blessing, *Database of cavitation effects in nozzles for model verification - Geometry and pressure effects on cavitating nozzle flow*, in *I-LEVEL Confidential report*. . Daimlerchrysler AG, 2002.

- 
218. Soteriou, C., R.J. Andrews, and M. Smith, *Direct Injection Diesel Sprays and the Effect of Cavitation and Hydraulic Flip on Atomization*. SAE Paper 950080, 1995.
  219. Petropoulou, S., M. Gavaises, and A. Theodorakakos, *An Adjoint Method for Controlled Cavitation Inverse Nozzle Design*, in *13<sup>th</sup> IPC conference*: Gyeongju, Korea, 2005.
  220. Petropoulou, S., M. Gavaises, and A. Theodorakakos, *An Adjoint method for controlled cavitation Inverse nozzle design*. Int. J. of Automotive Technology, 7(3): pp. 283-288, 2006.
  221. Petropoulou, S., M. Gavaises, and A. Theodorakakos, *An Adjoint Method for Hole Cavitating Control Through Inverse Nozzle Design*. SAE Paper 06P-162, 2006.
  222. *Catepillar Fuel Systems, Cavitation Effect and Solutions*. Private Communication, 2006.